



Virtual Tape Control System

Installation and Configuration Guide

Version 5.1.0

CRC Update Only

Proprietary Information Statement

This document and its contents are proprietary to Storage Technology Corporation and may be used only under the terms of the product license or nondisclosure agreement. The information in this document, including any associated software program, may not be reproduced, disclosed or distributed in any manner without the written consent of Storage Technology Corporation.

Limitations on Warranties and Liability

This document neither extends nor creates warranties of any nature, expressed or implied. Storage Technology Corporation cannot accept any responsibility for your use of the information in this document or for your use of any associated software program. You are responsible for backing up your data. You should be careful to ensure that your use of the information complies with all applicable laws, rules, and regulations of the jurisdictions in which it is used.

Warning: No part or portion of this document may be reproduced in any manner or in any form without the written permission of Storage Technology Corporation.

Restricted Rights

Use, duplication, or disclosure by the U.S. Government is subject to restrictions as set forth in subparagraph (c) (1) (ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227–7013 or subparagraphs (c) (1) and (2) of the Commercial Computer Software — Restricted Rights at 48 CFR 52.227–19, as applicable.

Export Destination Control Statement

These commodities, technology or software were exported from the United States in accordance with the Export Administration Regulations. Diversion contrary to U.S. law is prohibited.

First Edition, Revision G - November, 2003

CRC Update Only

This edition applies to Version 5.1.0 of the Virtual Tape Control System software. Information in this publication is subject to change. Send comments about this publication to:

Storage Technology Corporation
Manager, Software Information Development
One StorageTek Drive
Louisville, Colorado 80028-5209

OR

E-mail us at: sid@stortek.com

© 2003 Storage Technology Corporation. All rights reserved. StorageTek, the StorageTek logo and the following are trademarks or registered trademarks of Storage Technology Corporation:

StorageTek®
Nearline®
Virtual Storage Manager (VSM)[™]
Expert Library Manager (ExLM)[™]
Expert Performance Reporter (ExPR)[™]
Host Software Component (HSC)[™]
TimberLine[™]

Other products and names mentioned herein are for identification purposes only and may be trademarks of their respective companies.

About this Book

Virtual Tape Control System 5.1.0 (VTCS 5.1.0, hereafter referred to as “VTCS”) is MVS host software, which together the portions of NCS 5.1.0 that support VTCS and the Virtual Tape Storage Subsystem (VTSS), comprises Virtual Storage Manager (VSM).

Audience

This guide is for StorageTek or customer personnel who are responsible for installing configuring VTCS and VSM. See *VTCS Command and Utility Reference* for information about the following:

- VTCS and NCS (virtual) commands and utilities
- HSC SMF records for VTCS
- VTD commands

Also see: *VTCS Administrator's Guide* for information about VTCS administration tasks.

Reader's Comments

If you have comments on this book, please e-mail us at sid@stortek.com and include the document title and number with your comments.

Prerequisites

To perform the tasks described in this guide, you should already understand the following:

- MVS or OS/390 operating system
- JES2 or JES3
- System Management Facility (SMF)
- System Modification Program Extended (SMP/E)
- Nearline Control Solution (NCS)

About the Software

This guide applies to VTCS 5.1.0 and NCS 5.1.0 and above. VTCS executes in the native MVS or OS390 environment and does not use or require OS390 OpenEdition services.

How this Guide is Organized

This guide contains the following sections:

- Chapter 1 “Planning for Installation”
- Chapter 2 “Preparing for Installation”
- Chapter 3 “Installing HSC and VTCS”
- Chapter 4 “Reconfiguring NCS”
- Chapter 5 “Configuring VSM”
- Appendix A “VSM Configuration Record”
- Appendix C “VSM2 and VSM3 Logical Pathing”
- Appendix D “VSM4 Configuration”
- “Glossary”
- “Index”

What's New in This Guide?

First Edition, Revision G

The First Edition, Revision G of this guide contains technical updates and corrections. Specifically, the PTF numbers for T9840C/T9940B support **have changed** as shown in Table 1.

First Edition, Revision F

The First Edition, Revision F of this guide contains technical updates and corrections.

First Edition, Revision E

The First Edition, Revision E of this guide contains the updates described in Table 1.

Table 1. Updates to VTCS Installation and Configuration Guide, First Edition, Revision E

This Enhancement...	...is described in...	...and is available via these PTFs...
<ul style="list-style-type: none"> • T9840C support • T9940B support 	<ul style="list-style-type: none"> • “Special VTCS Considerations for T9840C Media” on page 21 • “Special VTCS Considerations for T9940B Media” on page 20 	<ul style="list-style-type: none"> • L1H11PU for VTCS 5.1 (SWS5100) • L1H11PT for HSC 5.1 (SOS5100)



Note: T9840C and T9940B support is available **only** for VSM3s and VSM4, **not** for VSM2s.

First Edition, Revision D

The First Edition, Revision D of this guide contains the updates described in Table 5.

Table 2. Updates to VTCS Installation and Configuration Guide, First Edition, Revision D

This Enhancement...	...is described in...	...and is available via these PTFs...
Updates to LibraryStation for 256 VTD support for VSM4	“MVS/CSC and Non-MVS/CSC Client Connection Values” on page 26	L1S1038 for LibraryStation 5.1

First Edition, Revision C

The First Edition, Revision C, contains technical updates and corrections.

**First Edition,
Revision B**

The newest generation VTSS is the VSM4, which provides the following advantages over its predecessors:

- Enhanced connectivity options.
- Greater throughput.
- Greater VTSS capacity.
- 4x the number of VTDs and 3x the maximum number of VTVs per VTSS.
- Improved reliability and serviceability.

Table 3 summarizes the VSM3 to VSM4 enhancements that you see from a software and system configuration perspective.

Table 3. VSM3 to VSM4 Comparison: Software and System Configuration Enhancements

Product Feature	VSM3	VSM4
ESCON Interfaces	16 total where: <ul style="list-style-type: none"> • 2 to 14 can be host channels • 2 to 8 can be Nearlink/CLINK connections 	32 total where: <ul style="list-style-type: none"> • 2 to 28 can be host channels • 2 to 16 can be Nearlink/CLINK connections <p>Note: On a VSM4, each ICE3 ESCON interface card contains two pairs of ESCON ports. Each pair is controlled by its own Channel Interface Processor (CIP). Each CIP switches between the two ports, so that only one port can transfer data at a time.</p>
Maximum Logical Paths	128	512 Note: VSM4 provides a theoretical maximum of 512 logical paths per VTSS, but you cannot allocate all 512 logical paths for host-to-VTSS connections.
VTDs per VTSS	64	256
Maximum resident VTVs per VTSS	100,000	300,000

The First Edition, Revision B of this guide contains the updates for VSM4 support described in Table 4.

Table 4. Updates to VTCS Installation and Configuration Guide for VSM4 Support, First Edition, Revision B

This Enhancement...	...is described in...
VSM4 support	<ul style="list-style-type: none"> Table 7 on page 5 “Converting the Formatted CDS to VSM Extended Format” on page 71 Appendix D “VSM4 Configuration”

First Edition, Revision A

The First Edition, Revision A contains technical updates and corrections.

First Edition

The *VTCS Installation and Configuration Guide* is a new book for VTCS 5.1, and consists of Chapters 2 through 6 plus Appendix A of the *VTCS 5.0 Installation, Configuration, and Administration Guide*. The First Edition of this guide contains the updates described in Table 5.

Table 5. Updates to VTCS Installation and Configuration Guide, First Edition

This Enhancement...	...is described in...
Updates to HSC SLUCONDB to support VTCS DELSCR(YES)	“Deleting Scratched VTVs” on page 42
Enhancements to Clustered VTSS configurations	“Clustered VTSS Requirements” on page 8
Updates to StorageTek MVS Client (SMC)	<ul style="list-style-type: none"> “VTCS System Software Requirements” on page 5 “VSM Esoterics and Esoteric Substitution” on page 12 “SMC Installation Considerations” on page 60 “Routing Data Sets to VSM” on page 93
HSC Extended CDS	“Converting the Formatted CDS to VSM Extended Format” on page 71
ANSI label support, which allows non-MVS/CSC clients, such as the Unisys CSC, to use ANSI labels for VTVs.	<ul style="list-style-type: none"> “Connecting Non-MVS/CSC Clients to VSM” on page 86. In this procedure, you define an HSC subpool from which the client requests scratch VTV mounts. If you define this subpool as ANSI label, VTCS will mount scratch VTV as ANSI labeled tapes from this subpool.

Conventions for Reader Usability

Conventions are used to shorten and clarify explanations and examples within this book.

Typographic

The following typographical conventions are used in this book:

- **Bold** is used to introduce new or unfamiliar terminology.
- Letter Gothic is used to indicate command names, filenames, and literal output by the computer.
- **Letter Gothic Bold** is used to indicate literal input to the computer.
- *Letter Gothic Italic* is used to indicate that you must substitute the actual value for a command parameter. In the following example, you would substitute your name for the “username” parameter.
- Logon *username*
- A bar (|) is used to separate alternative parameter values. In the example shown below either username or systemname must be entered.
- Logon *username|systemname*
- Brackets [] are used to indicate that a command parameter is optional.
- Ellipses (...) are used to indicate that a command may be repeated multiple times.
- The use of mixed upper and lower case characters (for non–case sensitive commands) indicates that lower case letters may be omitted to form abbreviations. For example, you may simply enter **Q** when executing the **Quit** command.

Keys

Single keystrokes are represented by double brackets [[]] surrounding the key name. For example, press [[ESC]] indicates that you should press only the escape key.

Combined keystrokes use double brackets and the plus sign (+). The double brackets surround the key names and the plus sign is used to add the second keystroke. For example, press [[AL]] + [[C]] indicates that you should press the alternate key and the C key simultaneously.

Enter Command

The instruction to “press the [[ENTER]] key” is omitted from most examples, definitions, and explanations in this book.

For example, if the instructions asked you to “enter” **Logon pat**, you would type in **Logon pat** and press lENTERm.

However, if the instructions asked you to “type” **Logon pat**, you would type in **Logon pat** and you would *not* press [[ENTER]].

Symbols

The following symbols are used to highlight text in this book.



Warning: Information necessary to keep you from damaging your hardware or software.



Caution: Information necessary to keep you from corrupting your data.

Hint: Information that can be used to shorten or simplify your task or they may simply be used as a reminder.

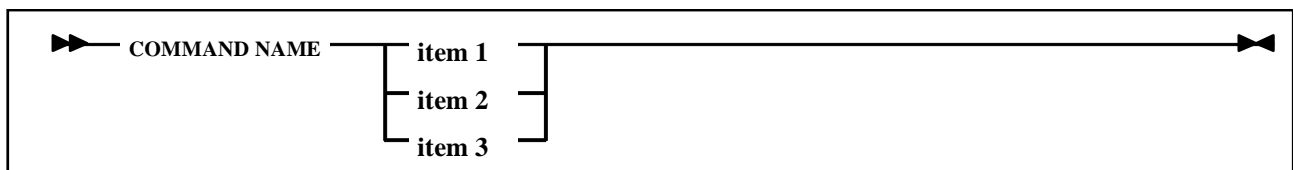


Note: Information that may be of special interest to you. Notes are also used to point out exceptions to rules or procedures.

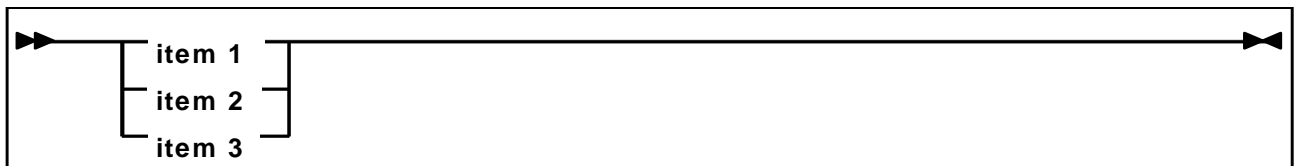
Syntax

Syntax flow diagram conventions include the following:

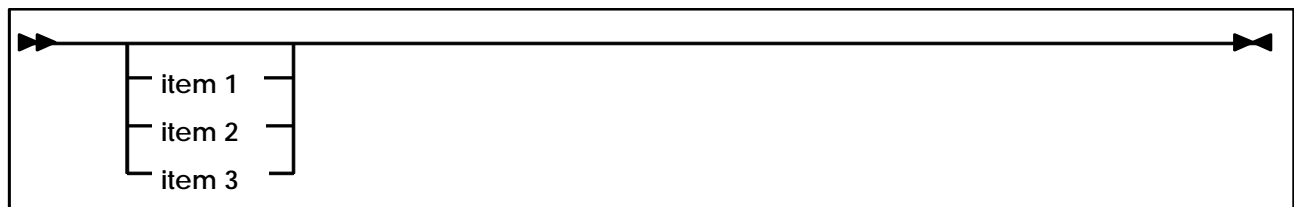
Flow Lines—Syntax diagrams consist of a horizontal baseline, horizontal and vertical branch lines and the command text. Diagrams are read left to right and top to bottom. Arrows show flow and direction.



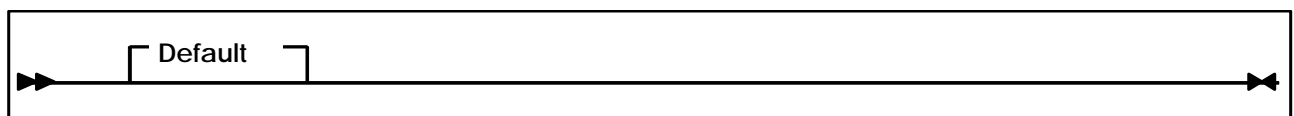
Single Required Choice—Branch lines (without repeat arrows) indicate that a single choice must be made. If one of the items to choose from is on the baseline of the diagram, one item must be selected.



Single Optional Choice—If the first item is on the line below the baseline, one item may optionally be selected.



Defaults—Default values and parameters appear above the baseline.



Repeat Symbol—A repeat symbol indicates that more than one choice can be made or that a single choice can be made more than once. The repeat symbol shown in the following example indicates that a comma is required as the repeat separator.



Keywords—All command keywords are shown in all upper case or in mixed case. When commands are not case sensitive, mixed case implies that the lowercase letters may be omitted to form an abbreviation.

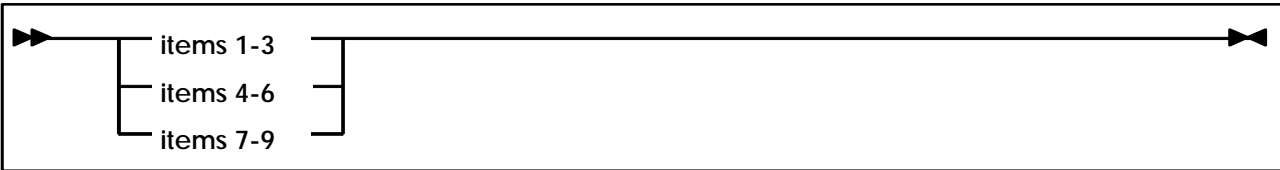
Variables—Italic type is used to indicate a variable.

Alternatives—A bar (|) is used to separate alternative parameter values.

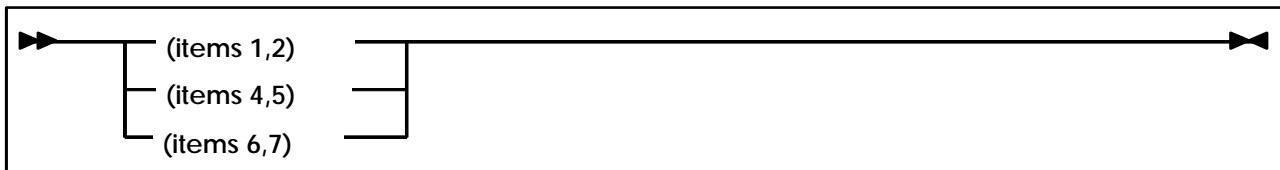
Optional—Brackets [] are used to indicate that a command parameter is optional.

Delimiters—If a comma (,), a semicolon (;), or other delimiter is shown with an element of the syntax diagram, it must be entered as part of the statement or command.

Ranges—An inclusive range is indicated by a pair of elements of the same length and data type, joined by a dash. The first element must be strictly less than the second element.



Lists—A list consists of one or more elements. If more than one element is specified, the elements must be separated by a comma or a blank and the entire line must be enclosed by parentheses.



Related Publications

The following publications provide additional information about VSM and StorageTek's Automated Cartridge System software and hardware.

VTCS and VSM

The VTCS and VSM documentation set consists of the following:

- *Introduction to VSM*, which you can request from your StorageTek representative
- The VTCS 5.1.0 Information CD-ROM, which contains PDF file formats of *Virtual Tape Control System Installation and Configuration Guide*, *Virtual Tape Control System Command and Utility Reference*, *Virtual Tape Control System Administrator's Guide*, *Virtual Tape Control System Messages*, and *Virtual Tape Control System XML Reference*
- *Virtual Tape Control System Installation and Configuration Guide* (this book)
- *Virtual Tape Control System Command and Utility Reference*
- *Virtual Tape Control System Administrator's Guide*
- *Virtual Tape Control System Messages*
- *Virtual Tape Control System Quick Reference*
- *Virtual Tape Control System XML Reference*
- *VSM Offsite Vault Disaster Recovery Guide* (supplied with the VSM Offsite Vault Disaster Recovery Feature)

VTSS

- *Virtual Storage Manager Planning, Implementation, and Usage Guide*
- *Virtual Storage Manager Physical Planning Guide*
- *VTSS Installation Guide*

NCS

- *NCS Installation Guide*
- *SMC Administration and Configuration Guide*

HSC-MVS Environment

- *Configuration Guide*
- *Operator's Guide*
- *System Programmer's Guide*
- *Messages and Codes*
- *System Programmer's Reference Summary*
- *Operator's Reference Summary*

LibraryStation

- *Configuration Guide*
- *Operator and System Programmer's Guide*
- *Messages and Codes*

MVS/CSC

- *Configuration Guide*
- *Operator Guide*
- *System Programmer Guide*
- *Messages and Codes*

ExPR

- *Introduction to ExPR*
- *ExPR SMP/E Installation*
- *ExPR MVS Configuration*
- *ExPR MVS Reports*
- *ExPR MVS Reference*

ExLM 4.0.0

The ExLM 4.0.0 documentation set consists of the following:

- The ExLM 4.0.0 Information CD-ROM, which contains PDF file formats of *ExLM Installation Guide*, *ExLM System Administrator's Guide*, *ExLM System Administrator's Guide - Field Tables Supplement*, and *ExLM Messages and Codes*
- *ExLM Installation Guide*
- *ExLM System Administrator's Guide*
- *ExLM System Administrator's Guide - Field Tables Supplement*
- *ExLM Messages and Codes*
- *ExLM Quick Reference*

ExLM 5.0.0

The ExLM 5.0.0 documentation set consists of the following:

- The ExLM 5.0.0 Information CD-ROM, which contains PDF file formats of the ExLM publications
- *ExLM Installation Guide*
- *ExLM System Administrator's Guide*
- *ExLM Messages and Codes*
- *ExLM Quick Reference* (includes information formerly provided in the *ExLM 4.0.0 System Administrator's Guide - Field Tables Supplement*)

IBM Publications

- *IBM ESA/390 Common I/O-Device Commands and Self Description*
- *IBM 3490 Magnetic Tape Subsystem*
Models A01, A02, A10, A20, B02, B04, B20, and B40
Introduction
- *IBM 3490 Magnetic Tape Subsystem*
Models A01, A02, A10, A20, B02, B04, B20, and B40
Hardware Reference
(Referred to in this book as the *IBM 3490 Hardware Reference*)
- *IBM 3490 Command Reference*
- *IBM 3480 Magnetic Tape Subsystem Reference*
- *IBM 3480 Installation Guide and Reference*
- *OS/390 V2R4.0 MVS Planning: Global Resource Serialization*
- *MVS Authorized Assembler Services Guide*

Online Documentation on the StorageTek CRC

The StorageTek Customer Resource Center (CRC) on the World Wide Web provides online versions in PDF format of this book, the related StorageTek publications listed on page xi, and many other StorageTek software and hardware publications.



To access PDF documents on the StorageTek CRC:

1. **Using an Internet browser such as Netscape or Internet Explorer, go to the StorageTek CRC at:**

<http://www.support.storagetek.com/>

2. **Click the Login link.**
3. **Fill in the login information.**

If this is the first time you have used the CRC, click Request a CRC password and fill in the requested information. You should receive your account information within two business days.

4. **From the upper left bar, click Product Information and Current Products from the dropdown links.**
5. **Select Software from the Product Family dropdown menu and click Next.**

Click the desired product link from the Product Categories and navigate to the documents you want to view.

Technical Support

Refer to *Requesting Help from Software Support* for information about contacting StorageTek for technical support and for requesting changes to software products.

Document Effectivity

EC Number	Date	Doc Kit Number	Type	Effectivity
128582	December 2002	---	First Edition	This document applies to VTCS, Version 5.1.0.
---	May 2003	---	First Edition, Revision A	This document applies to VTCS, Version 5.1.0.
---	June 2003	---	First Edition, Revision B	This document applies to VTCS, Version 5.1.0.
---	August 2003	---	First Edition, Revision C	This document applies to VTCS, Version 5.1.0.
---	September 2003	---	First Edition, Revision D	This document applies to VTCS, Version 5.1.0.
---	October 2003	---	First Edition, Revision E	This document applies to VTCS, Version 5.1.0.
---	November 2003	---	First Edition, Revision F	This document applies to VTCS, Version 5.1.0.
---	November 2003	---	First Edition, Revision G	This document applies to VTCS, Version 5.1.0.

Contents

About this Book	iii
Audience	iii
Reader's Comments	iii
Prerequisites	iii
About the Software	iii
How this Guide is Organized	iv
What's New in This Guide?	v
First Edition, Revision G	v
First Edition, Revision F	v
First Edition, Revision E	v
First Edition, Revision D	v
First Edition, Revision C	v
First Edition, Revision B	vi
First Edition, Revision A	vii
First Edition	vii
Conventions for Reader Usability	viii
Typographic	viii
Keys	viii
Enter Command	viii
Symbols	ix
Syntax	ix
Related Publications	xi
VTCS and VSM	xi
VTSS	xi
NCS	xii
ExPR	xiii
ExLM 4.0.0	xiii
ExLM 5.0.0	xiii
IBM Publications	xiv
Online Documentation on the StorageTek CRC	xv
Technical Support	xv
Document Effectivity	xvii

Chapter 1. Planning for Installation	1
VTCS Installation Summary and Checklist	2
Verifying VSM Software and Hardware Prerequisites	4
VTCS System Software Requirements	5
Nearline Hardware Requirements	7
Clustered VTSS Requirements	8
Determining VSM Configuration Values	9
HSC Definition Data Set Names	9
VTSS Identifiers	10
VTD Unit Addresses	11
VSM Esoterics and Esoteric Substitution	12
VTV Definitions	17
RTD Unit Addresses and Identifiers	19
MVC Definitions	20
MVS/CSC and Non-MVS/CSC Client Connection Values	26
HSC CDS DASD Space	38
Tape Management System DASD Space	38
VSM Candidate Data Sets	38
HSC COMMPATH METHOD Value	38
Data Chaining a VTD Read Forward or Write Command	38
VSM Policies	39
Chapter 2. Preparing for Installation	53
Defining A Security System User ID for HSC, SMC, and VTCS	54
Configuring MVS Device Numbers and Esoterics	54
Assigning MVS Device Numbers to VTDs	54
Associating VTD MVS Device Numbers and Esoterics	54
Assigning MVS Device Numbers to RTDs	54
Setting the MVS Missing Interrupt Handler (MIH) Value	55
Specifying the Region Size	55

Chapter 3. Installing HSC and VTCS	57
Reviewing Coexistence Requirements	58
Verifying Installation Materials	58
VTCS Installation Tape Contents	59
VTCS FMIDs.	60
Installing HSC and SMC	60
SMC Installation Considerations	60
Receiving the VTCS 5.1.0 FMID	61
Receiving the VTCS 5.1.0 Service	61
Creating the VTCS 5.1.0 LINKLIB Data Sets and Defining Libraries to the HSC Target Zone ..	62
Applying the VTCS 5.1.0 FMID	63
Accepting the VTCS 5.1.0 FMID	64
Applying the VTCS 5.1.0 Service	65
APF Authorizing the VTCS LINKLIB	65
Modifying the HSC Startup Procedure to include the VTCS 5.1.0 LINKLIB	65
Chapter 4. Reconfiguring NCS	67
Updating the HSC LIBGEN	68
Verifying the LIBGEN	70
Formatting the New CDS	70
Converting the Formatted CDS to VSM Extended Format.	71
Updating the HSC Definition Data Sets	73
Creating VTD UNITATTR Statements	73
Creating RTD UNITATTR Statements	73
Creating MVC VOLATTR Statements	73
Creating an MVC Pool	73
Creating and Using VSM Management and Storage Classes	74
Basic Procedure for Creating and Using VSM Management and Storage Classes	75
Using Storage and Management Classes to Group Multiple Workloads on Shared MVCs ..	76
Using Storage and Management Classes to Segregate Individual Workloads on	
Separate Sets of MVCs	77
Using Storage and Management Classes to Specify the Source and Target MVC for	
VTV Consolidation.	78
Creating and Using Named MVC Pools	79
Updating the HSC PARMLIB Member (SLSSYSxx)	82
Adding SMF Parameters for VTCS to SYS1.PARMLIB	83
Connecting MVS/CSC Clients to VSM	84
Connecting Non-MVS/CSC Clients to VSM	86
Restarting NCS	87

Chapter 5. Configuring VSM.	89
Defining VSM Security	90
Defining MVC Pool Volser Authority	90
Defining VTCS Command Authority	91
Configuring VTCS	91
Running the HSC MERGECDS Utility	92
Updating the Tape Management System	92
Updating HSM	93
Routing Data Sets to VSM	93
The StorageTek DFSMS Interface	94
HSC or MVS/CSC TAPERREQ Statements	95
HSC User Exits	95
MVS/CSC User Exits	95
Starting VTCS	96
Appendix A. VSM Configuration Record.	97
Appendix B. VSM Connectivity Requirements	101
Appendix C. VSM2 and VSM3 Logical Pathing	103
VSM2 and VSM3 Logical Pathing Overview	104
Host Paths for VTSS with 8 ICE Cards, 4 RTD Nearlink Connections	105
Host Paths for VTSS with 8 ICE Cards, 8 RTD Nearlink Connections	106
Host Paths for VTSS with 4 ICE Cards, 4 RTD Nearlink Connections	106
VSM2 and VSM3 Logical Path Planning and Configuration Example	107
Step 1: Determine Logical Pathing Requirements	108
Step 2: Determine Channel Requirements and Allocate Channels	112
Step 3: Allocate Logical Paths	113
Step 4: Code The IOCP	122

Appendix D. VSM4 Configuration.....	125
VSM4 with 32 Ports	127
VSM4 Configuration Examples - 32 Ports	131
VSM4 Configuration Example: 16 Host Ports, 16 RTD Ports	132
VSM 4 Configuration Example: 20 Host Ports, 12 RTD Ports	134
VSM4 with 16 Ports	136
VSM4 Configuration Examples - 16 Ports	139
VSM4 Configuration Example: 8 Host Ports, 8 RTD Ports	140
VSM 4 Configuration Example: 10 Host Ports, 6 RTD Ports	142
IOCP Example for Single MVS Host Connected to a VSM4 Via ESCON Directors.....	144
Logical Paths for VSM 4 with 32 Ports.....	146
Glossary.....	147
Index.....	159

List of Figures

Figure 1. Example VSM Esoteric Structure for JES3 with Tape SETUP.	15
Figure 2. JCL Example: Receiving the VTCS 5.1.0 FMID	61
Figure 3. JCL Example: Receiving the VTCS 5.1.0 service	61
Figure 4. JCL Example: Creating the SWSLINK data sets and defining libraries to the HSC target zone	62
Figure 5. JCL Example: Applying the VTCS 5.1.0 FMID	63
Figure 6. JCL Example: Accepting the VTCS 5.1.0 FMID	64
Figure 7. JCL Example: Applying the VTCS 5.1.0 service	65
Figure 8. JCL Example: Modifying the HSC started task to include the SWSLINK library	65
Figure 9. CDS Format Compatibility	71
Figure 10. CONFIG example: VER(NEW) specified to prepare to convert the CDS to VSM Extended Format	72
Figure 11. MERGEcds example: converting the CDS to VSM Extended Format	72
Figure 12. Storage Classes for Workload Grouping	76
Figure 13. Management Classes for Workload Grouping.	76
Figure 14. TAPEREQ Statements for Workload Grouping	76
Figure 15. Storage Classes for Workload Segregation	77
Figure 16. Management Classes for Workload Segregation	77
Figure 17. TAPEREQ Statements for Workload Segregation	77
Figure 18. Storage Classes for Source and Target MVC for Consolidation	78
Figure 19. Management Class for Source and Target MVC for Consolidation	78
Figure 20. TAPEREQ Statement for Source and Target MVC for Consolidation	78
Figure 21. CONSOLID Utility JCL Specifying Source and Target MVC for Consolidation by Management Class	78
Figure 22. Example: Updating the HSC PARMLIB Member for VSM.	82
Figure 23. SYS1.PARMLIB member SMFPRMxx example for VTCS SMF records.	83
Figure 24. Example RACF MVC volser access file	90
Figure 25. Example RACF VTCS command authorization file	91
Figure 26. Full VSM Connectivity	101
Figure 27. Partial VSM Connectivity	102
Figure 28. PARTITION Parameter on the CHPID Statement-Access List	123
Figure 29. PARTITION Parameter on the CHPID Statement-Candidate List	123
Figure 30. IOCP Example for VTSS01.	124
Figure 31. VSM4 with 32 Ports.	127
Figure 32. VSM4 with 16 Host Ports, 16 RTD Ports	132
Figure 33. CONFIG example: VSM4 with 16 Host Ports, 16 RTD Ports	133
Figure 34. VSM4 with 20 Host Ports, 12 RTD Ports	134
Figure 35. CONFIG example: VSM4 with 20 Host Ports, 12 RTD Ports	135
Figure 36. VSM4 with 16 Ports.	136
Figure 37. VSM4 with 8 Host Ports, 8 RTD Ports	140
Figure 38. CONFIG example: VSM4 with 8 Host Ports, 8 RTD Ports	141
Figure 39. VSM4 with 10 Host Ports, 6 RTD Ports	142
Figure 40. CONFIG example: VSM4 with 10 Host Ports, 6 RTD Ports	143
Figure 41. Configuration Diagram: Single MVS Host Connected to a VSM4 via ESCON Directors. .	144
Figure 42. IOCP Example: Single MVS Host Connected to a VSM4 via ESCON Directors	145
Figure 43. Logical Paths for VSM 4 with 32 Ports, 31 Hosts, 16 RTDs	146

List of Tables

Table 1. Updates to VTCS Installation and Configuration Guide, First Edition, Revision E.	v
Table 2. Updates to VTCS Installation and Configuration Guide, First Edition, Revision D.	v
Table 3. VSM3 to VSM4 Comparison: Software and System Configuration Enhancements.	vi
Table 4. Updates to VTCS Installation and Configuration Guide for VSM4 Support, First Edition, Revision B.	vii
Table 5. Updates to VTCS Installation and Configuration Guide, First Edition.	vii
Table 6. VTCS Installation Summary and Checklist.	2
Table 7. VTCS 5.1.0 Minimum Software Requirements.	5
Table 8. Third Party Tape Copy Software for Migrating Data to VSM.	6
Table 9. VSM Nearline Hardware Requirements.	7
Table 10. Clustered VTSS Requirements.	8
Table 11. LibraryStation VSM2/3 Virtual ACS Locations for VTDs.	26
Table 12. LibraryStation VSM4 Virtual ACS Locations for VTDs.	29
Table 13. VTCS 5.1.0 Product Tape Contents.	59
Table 14. VTCS 5.1.0 Service Tape Contents.	59
Table 15. Security Class, Resource Class, and Access Values for MVC Pool Volser Authority.	90
Table 16. VSM Configuration Record.	97
Table 17. Host Paths for VTSS with 8 ICE Cards, 4 RTD Nearlink Connections.	105
Table 18. Host Paths for VTSS with 8 ICE Cards, 8 RTD Nearlink Connections.	106
Table 19. Host Paths for VTSS with 4 ICE Cards, 4 RTD Nearlink Connections.	106
Table 20. Logical Pathing Requirements for VTSS01.	110
Table 21. Logical Pathing Requirements for VTSS02.	111
Table 22. Channel Requirements for Each CPU.	112
Table 23. Channel Allocation for Each CPU.	112
Table 24. VTSS01 Host Logical Paths for Throughput.	113
Table 25. VTSS01 Host Logical Paths for Throughput and Connectivity.	114
Table 26. VTSS01 Host Logical Paths for Throughput, Connectivity, and Redundancy.	115
Table 27. VTSS01 Host Logical Paths for Throughput, Connectivity, and Redundancy, and RTD Connections.	116
Table 28. VTSS01 Unallocated Host Logical Paths.	117
Table 29. VTSS02 Host Logical Paths for Throughput.	118
Table 30. VTSS02 Host Logical Paths for Throughput and Connectivity.	118
Table 31. VTSS02 Host Logical Paths for Throughput, Connectivity, and Redundancy.	119
Table 32. VTSS02 Host Logical Paths for Throughput, Connectivity, and Redundancy, and RTD Connections.	120
Table 33. VTSS02 Unallocated Host Logical Paths.	121
Table 34. VSM3 to VSM4 Comparison: Software and System Configuration Enhancements.	125
Table 35. Optimizing VSM4 Port Operations.	128
Table 36. VSM\$8 Configuration Options - 32 Ports.	130
Table 37. VSM4 Configuration Options - 16 Ports.	137

Chapter 1. Planning for Installation

Before doing the tasks described in Chapter 2 “Preparing for Installation”, complete the planning tasks described in the following sections:

- “Verifying VSM Software and Hardware Prerequisites” on page 4.
- “Determining VSM Configuration Values” on page 9. Use Table 16 on page 97 to record these VSM configuration values, which you use to complete the tasks in this guide.

Use “VTCS Installation Summary and Checklist” on page 2 to help plan and verify completion of your system’s installation and configuration tasks.



Hint: Your StorageTek representatives will help you plan and install the VTSS hardware. If you are adding Nearline hardware, such as LSMs or transports to be used as RTDs, StorageTek will also help you install and configure this hardware.

To plan the VTSS configuration and install and configure your VTSS hardware, StorageTek will use the VTSS publications on page xi and the VSM pre-sales planning tool. This tool produces data to help define the optimum VSM solution for your business needs and to identify VSM candidate data sets.

Your StorageTek representatives will also help coordinate the installation and configuration of the VSM hardware with the software installation and configuration tasks described in this guide. For example, before you can verify the updated HSC LIBGEN as described in “Verifying the LIBGEN” on page 70, if your system’s RTDs are new transports and you will share them between VSM and MVS, you must install them and define their MVS unit addresses via the HCD facility.

VTCS Installation Summary and Checklist

Use the checklist in Table 6 to help plan and verify completion of your system's installation and configuration tasks.

Table 6. VTCS Installation Summary and Checklist

Task	✓ to Verify Completion
Planning for Installation	
“Verifying VSM Software and Hardware Prerequisites” on page 4	
“Determining VSM Configuration Values” on page 9	
Preparing for Installation	
“Defining A Security System User ID for HSC, SMC, and VTCS” on page 54	
“Configuring MVS Device Numbers and Esoterics” on page 54	
“Setting the MVS Missing Interrupt Handler (MIH) Value” on page 55	
“Specifying the Region Size” on page 55	
Installing VTCS 5.1.0	
“Reviewing Coexistence Requirements” on page 58	
“Verifying Installation Materials” on page 58	
“Installing HSC and SMC” on page 60	
“If you are running at an SMP/E level lower than that supplied with OS/390 Version 2.5, you must accept the HSC FMID (S0S5100 for HSC 5.1.0) and all HSC maintenance before installing VTCS 5.1.0.” on page 60	
“Receiving the VTCS 5.1.0 FMID” on page 61	
“Receiving the VTCS 5.1.0 Service” on page 61	
“Creating the VTCS 5.1.0 LINKLIB Data Sets and Defining Libraries to the HSC Target Zone” on page 62	
“Applying the VTCS 5.1.0 FMID” on page 63	
“Accepting the VTCS 5.1.0 FMID” on page 64	
“Applying the VTCS 5.1.0 Service” on page 65	
“APF Authorizing the VTCS LINKLIB” on page 65	
“Modifying the HSC Startup Procedure to include the VTCS 5.1.0 LINKLIB” on page 65	
Reconfiguring NCS	
“Updating the HSC LIBGEN” on page 68	

“Verifying the LIBGEN” on page 70	
“Formatting the New CDS” on page 70	
“Converting the Formatted CDS to VSM Extended Format” on page 71	
“Updating the HSC Definition Data Sets” on page 73	
“Creating and Using VSM Management and Storage Classes” on page 74	
“Updating the HSC PARMLIB Member (SLSSYSxx)” on page 82	
“Connecting MVS/CSC Clients to VSM” on page 84	
“Connecting Non-MVS/CSC Clients to VSM” on page 86	
“Restarting NCS” on page 87	
Configuring VSM	
“Defining VSM Security” on page 90	
“Configuring VTCS” on page 91	
“Running the HSC MERGECDs Utility” on page 92	
“Updating the Tape Management System” on page 92	
“Updating HSM” on page 93	
“Routing Data Sets to VSM” on page 93	
“Starting VTCS” on page 96	

Verifying VSM Software and Hardware Prerequisites

VTCS System Software Requirements

Verify the software prerequisites for VTCS 5.1.0 listed in Table 7.

Table 7. VTCS 5.1.0 Minimum Software Requirements

Software Description	Minimum Version/Release
Operating System	<p>MVS 5.2.2 and above</p> <p>Note: NCS SMC JES3 requires JES3 5.1.1 or higher</p> <p>All versions of OS390</p>
Nearline Control Solution	<p>NCS 5.1</p> <p>Note:</p> <ul style="list-style-type: none"> VTCS 5.1.0 requires HSC 5.1.0 and will not run with previous versions of HSC. You can use the HSC 5.0 and above MERGEcds utility to convert the CDS to VSM Extended Format. The VSM Extended Format CDS is required for VTCS 5.1.0 or higher with VSM4s that are defined with greater than 64 VTDs and/or greater than 8 RTDs. VSM4 support is provided in base VTCS 5.1.0, and no additional PTFs are required for device support. However, StorageTek recommends you apply, as needed, the most current HSC and VTCS 5.1 maintenance (available on the StorageTek CRC). <p>Note that after you convert the CDS to VSM Extended Format, you cannot run VTCS 4.0.0 or lower against the converted CDS.</p> <p>For more information, see “Converting the Formatted CDS to VSM Extended Format” on page 71.</p> <ul style="list-style-type: none"> If you are using RMM in an MVS/CSC environment, MVS/CSC can share the tape management catalog with the host(s) running HSC if you have the following installed: <ul style="list-style-type: none"> RMM APAR OA03368 VSM3 microcode N01.00.65 or later or VSM4 microcode D01.00.03 or later. <p>Otherwise, the tape management catalog cannot be shared or VTV scratch mounts will fail.</p>

Expert Performance Reporter (optional software)	ExPR 4.0
Expert Library Manager (optional software)	<p>To use Expert Library Manager (ExLM) with VSM for VTV consolidation using ExLM, ExLM 4.0, HSC 4.0.0, and VTCS 4.0.0.</p> <p>For more information about using ExLM with VSM, see “Using ExLM to Manage VSM Resources” in Chapter 2 of the <i>ExLM System Administrator’s Guide</i>.</p>



Caution: In a VSM configuration with multiple hosts that share the same HSC CDS, StorageTek strongly recommends that you:

- Install VTCS on all MVS hosts by completing all tasks described in “Installing HSC and VTCS” on page 57. Installing VTCS on all MVS hosts ensures that these hosts cannot scratch an MVC.

Note that CONFIG lets you define MVS hosts that are not connected to a VTSS.

- Do *not* use a host that does not have VTCS installed to enter MVCs into an ACS, otherwise these MVCs will be eligible for selection as scratch volumes by any host in the configuration with HSC installed.

Third Party Tape Copy Software for Migrating Data to VSM

Table 8 lists Third Party tape copy software for migrating data to VSM.

Table 8. Third Party Tape Copy Software for Migrating Data to VSM

Product Name	Vendor
Beta55	Beta Systems Software AG
TelTape/390	Cartagena Software Limited
CA-1®/Copycat	Computer Associates International
CA-Dynam®/TLMS/Copycat	Computer Associates International
MediaMerge	eMag Solutions
FATSCopy	Innovation Data Processing
Tape/Copy	OpenTech Systems, Inc.
Zela	Software Engineering of America
CARTS-TS TapeSaver	UNICOM Systems, Inc.

Nearline Hardware Requirements

Verify the minimum Nearline hardware requirements listed in Table 9.

Table 9. VSM Nearline Hardware Requirements

Hardware	Requirement
LSMs	Any of the following but 9310 or 9740 recommended by StorageTek: 4410, 9310, 9740, and 9360
Transports and media	<p>VSM RTDs can be a mixture of 9490 (Timberline), 9490EE (Timberline EE), T9840A, T9840B, and T9940 transports. Each VTSS must have a minimum of two library-attached transports for each media type used for MVCs. For example, if your MVCs are STANDARD and ECART, you need a minimum of <i>either</i> two 9490s <i>or</i> two 9490EEs as RTDs. If your MVCs are STANDARD, ECART, ZCART, and STK1R, you need a minimum of two 9490EEs <i>and</i> two 9840s as RTDs.</p> <p>Note: Using 9840s as RTDs and using ZCART media with 9490EE RTDs requires HSC 2.1.0. You cannot use 9840s as RTDs, ZCART media, or 9740 LSMs in a configuration where HSC 2.1.0 and HSC 2.0.1 share the CDS.</p> <p>Using T9940s as RTDs requires HSC 4.0.0 and VTCS 4.0.0 with PTF L1H1043 applied.</p> <p>Valid media types for the supported RTDs are:</p> <p>9490: STANDARD, ECART</p> <p>9490EE: STANDARD, ECART, ZCART</p> <p>T9840A and T9840B: STK1R</p> <p>T9940: STK2, STK2P (both currently indicate 60 Gb cartridges)</p>

Clustered VTSS Requirements

Table 10. describes the requirements for implementing Clustered VTSS configurations.

Table 10. Clustered VTSS Requirements

Component	Requirement
Primary and Secondary VTSSs within a cluster	<p>The Primary and Secondary VTSSs can be any combination of VSM3 and VSM4 where the Secondary can be of any capacity. All hosts must be at VTCS 5.1.0 to enable this feature. For example, all of the following are valid:</p> <ul style="list-style-type: none"> • Primary VSM4, Secondary VSM3 • Primary VSM4, Secondary VSM4 • Primary VSM3, Secondary VSM3 • Primary VSM3, Secondary VSM4 (not recommended) <p>All hosts must be at VTCS 5.1.0 to enable this feature.</p>
Primary and Secondary VTSS microcode	<p>The Primary VTSS microcode must be at a level that supports sending replicated VTVS. The Secondary VTSS microcode must be at a level that supports receiving replicated VTVS and supports the use of the Secondary as a production VTSS. After the microcode is installed, the Clustering feature must be enabled at both the Primary and Secondary VTSS via an options floppy disk. See your StorageTek hardware service representative for details.</p>
VTCS software	<ul style="list-style-type: none"> • VTCS 5.1.0 (for enhanced clustered support) • The Advanced Management Feature (to enable the REPLICAT parameter of the MGMTclas statement)

Determining VSM Configuration Values

The following sections tell how to determine configuration values for your VSM system. Use Table 16. on page 97 to record these values. This table also provides a record of your site's VSM configuration, which can help you and StorageTek service troubleshoot problems with your VSM system.



Note: Unless otherwise noted, in each of the following sections, the values you determine must match wherever you use them. For example, the unit addresses described in “RTD Unit Addresses and Identifiers” on page 19 must match on the following:

- The HSC SLIDRIVS macro.
- The UNITATTR statement.
- If you will share these transports with MVS, when you assign MVS device addresses to these transports via the HCD facility.

HSC Definition Data Set Names

Determine the names of the data sets that will contain your VSM system's UNITATTR, VOLATTR, TAPEREQ, and MVCPool, MGMTclas and STORclas statements. MGMTclas and STORclas statements must reside in the same file (sequential data set or PDS member) for cross-validation.

VTSS Identifiers

Determine your system's 1 to 8 character VTSS identifiers, which you specify when you:

- Create UNITATTR statements to identify the VTDs available to each VTSS as described in “Creating VTD UNITATTR Statements” on page 73.
- Run VTCS CONFIG to initially install and configure your VSM system as described in “Configuring VTCS” on page 91.



Caution: Note the following:

- You specify the VTSS identifier *only* via the NAME parameter of the VTSS statement of CONFIG, which sets the VTSS identifier in both the VTSS microcode (as displayed in the Subsystem Name field in the LOP) and in the configuration area of the HSC CDS. After VSM is put into operation, the VTSS identifier is also stored in each VTV record in the CDS. Each VTV record contains the VTSS identifier on which that VTV is resident or, if the VTV is migrated, the VTV record contains the VTSS identifier from which the VTV was migrated.
- Once you set the VTSS identifier via the NAME parameter, you *cannot* change this identifier in the HSC CDS with CONFIG. CONFIG *will not* let you change the NAME parameter after an initial setting and changing the VTSS identifier using the Subsystem Name field of the LOP *cannot* change the VTSS identifier in the HSC CDS.
- You can, however, change a VTSS identifier when merging CDS.
- It is especially critical that you *do not* attempt to rename a VTSS that contains data on VTVs, which includes VTSS-resident VTVs and migrated VTVs!
- For an initial setting *only* (not a change), you can set the VTSS identifier in the NAME parameter only if the VTSS identifier value in the VTSS microcode is:
 - The factory setting (all blanks).
 - A value of 99999999 (eight 9s).

Therefore, for an initial setting *only*, if the name in the VTSS microcode *is not* all blanks or 99999999, your StorageTek hardware representative must use the VTSS LOP to set the VTSS identifier to 99999999 so you can set the VTSS identifier to the value you want via the NAME parameter.

VTD Unit Addresses

Determine MVS unit addresses for your system's VTDs as follows:

- For each VTSS in your VSM configuration, determine a unique unit address range for the VTDs in that VTSS. Do not use duplicate addresses or overlapping address ranges, either within the VTDs in a VTSS or across VTSSs.

You specify these unit addresses to create a separate UNITATTR statement for each VTSS as described in “Creating VTD UNITATTR Statements” on page 73.

- For each VTSS in your VSM configuration, you must also define its VTD unit addresses to VTCS via CONFIG. For each VTSS, use the *same* unit address range you specified on the corresponding UNITATTR statement for that VTSS.

In a multi-host, multi-VTSS configuration, you can configure your VTD unit addresses to restrict host access to VTSSs. Note that the VTVs created and MVCs initially written to from a VTSS are considered that VTSS's resources, so only hosts with access to a VTSS also have access to its VTVs and MVCs.

- For each HSC host, use the HCD facility to define to MVS the VTDs that host can access as described in “Assigning MVS Device Numbers to VTDs” on page 54. The unit addresses you specify via the HCD facility *must* match the unit address range you specified for that host via CONFIG.
- If you use MIM or GRS, add VTDs to the list of managed devices.

VSM Esoterics and Esoteric Substitution

Default SMC Device Allocation for VTDs in Multi-VTSS Systems

Where multiple VTSSs are eligible for allocation (from the UNITNAME in the JCL, SMS unit substitution, or from TAPEREQ statements), the **default** SMC device allocation for VTDs is as follows:

- For *specific* volume requests, if the VTV is VTSS-resident, SMC will restrict allocation to the VTDs in that VTSS.

If the VTV is *not* resident, SMC will restrict allocation to the VTDs in the VTSSs that have physical access to the MVC(s), where allocation will be restricted to a the VTDs of a single VTSS selected at random from the all the VTSSs that have physical access to the MVC(s).

- For *scratch* volume requests:
 - If a Management Class is assigned to the VTV, then SMC allocates a VTD that can best satisfy the Management Class policies. For example, if the Management Class specifies a migration policy via the MIGPOL parameter (ACS ID, MVC media, and duplexing), then SMC allocates a VTD by selecting a VTSS attached to ACS(s) that best satisfy the policy. Similarly, if the Management Class specifies VTV replication via the REPLICAT(YES) parameter, SMC attempts to select a VTD on the Primary VTSS of a full-function Cluster.
 - If a Management Class is **not** assigned to the VTV, then SMC attempts to allocate the VTD in a VTSS with the greatest difference between the DBU and HAMT.

The following sections describe the requirements for defining and using VSM esoterics to influence VTD allocation for the following interfaces:

- The StorageTek DFSMS interface
- HSC and MVS/CSC TAPEREQ statements and User Exits for the following environments:
 - JES2 and JES3 without tape SETUP
 - JES3 with tape SETUP



Note:

- To ensure that virtual requests, like, actually go to virtual, StorageTek recommends that esoterics contain **only** VTDs.
- For any esoteric that you design, you must define the esoteric and associate it with the MVS device numbers for the VTDs you have chosen for that esoteric; see “Associating VTD MVS Device Numbers and Esoterics” on page 54.

- Assigning a Management Class to a VTV **requires** allocating a VTD in a VTSS that can satisfy the requirements of the assigned Management Class. For any jobs that route data to VSM via esoteric substitution **and** assign a Management Class to the data, **ensure** that you specify an esoteric that includes VTSSs that can satisfy the requirements of the Management Class!
- As described in “Routing Data Sets to VSM” on page 93, you can also route data sets to VSM by:
 - Using TAPEREQ statements or HSC User Exits that do *not* use esoteric substitution
 - Changing existing JCL or writing new JCL

VSM Esoterics and Esoteric Substitution for the StorageTek DFSMS Interface

If you use these interfaces, you can use any valid esoteric defined to MVS. For these interfaces, begin your esoteric names with an alphabetic character to meet SMS requirements.

With these interfaces, for example, you can define and use any or all of the following:

- An esoteric that represents all the VTDs in your VSM system
- An esoteric for each VTSS that represents all the VTDs in that VTSS
- An esoteric that represents a subset of the VTDs in a single VTSS
- An esoteric that spans VTSSs



Note: The StorageTek DFSMS interface cannot assign Management Class to a VTV at VTD allocation.

VSM Esoterics and Esoteric Substitution for TAPEREQ Statements and User Exits

If you use the HSC or MVS/CSC TAPEREQ statement or User Exits, see the following sections for information on how to define and use VSM esoterics for these environments:

- JES2 and JES3 without tape SETUP
- JES3 with tape SETUP

VSM Esoterics for JES2 and JES3 without Tape SETUP. In JES2 and JES3 without tape SETUP, for TAPEREQ statements or User Exits, esoteric definition and substitution is as follows:

- Esoteric definition is **optional** with these interfaces in these environments. That is, if you do not use esoteric substitution for VSM, you do not have to define any VSM esoterics.
- As long as the esoteric is a valid esoteric defined to MVS, esoteric substitution works as follows:
 - Allocation determines the common drives between the specified esoteric and the Eligible Device List (EDL).
 - As long as there are sufficient drives in the list of common drives, then allocation continues using this list of common drives.
- If you use esoteric substitution a multi-VTSS environment, StorageTek recommends that you:
 - Define an esoteric for each VTSS.
 - Ensure that each VTSS esoteric represents exactly the entire range of devices for only that VTSS.
 - Ensure that the VTSS esoteric name matches the VTSS identifier on the VTD UNITATTR statement (which must match the VTSS identifier defined via the CONFIG utility).

This approach allows you to do esoteric substitution at the VTSS level and also allows you to use the same VTSS identifier defined in VTD UNITATTR statement and via the CONFIG utility.

- You can also define other virtual esoterics. For example, you can define and substitute an esoteric that represents all of the VTDs in all VTSSs or an esoteric that represents all VTSSs that comprise VTSS Clusters. For more information on how VTCS allocates VTDs in a multi-VTSS system, see “Default SMC Device Allocation for VTDs in Multi-VTSS Systems” on page 12.
- For consistency, especially in installations that run both JES2 and JES3, you may want to define an esoteric structure for JES2 such as shown for JES3 with tape SETUP in Figure 1 on page 15.

VSM Esoterics for JES3 with Tape SETUP

Figure 1 is an example of a VSM esoteric structure for JES3 with tape SETUP.

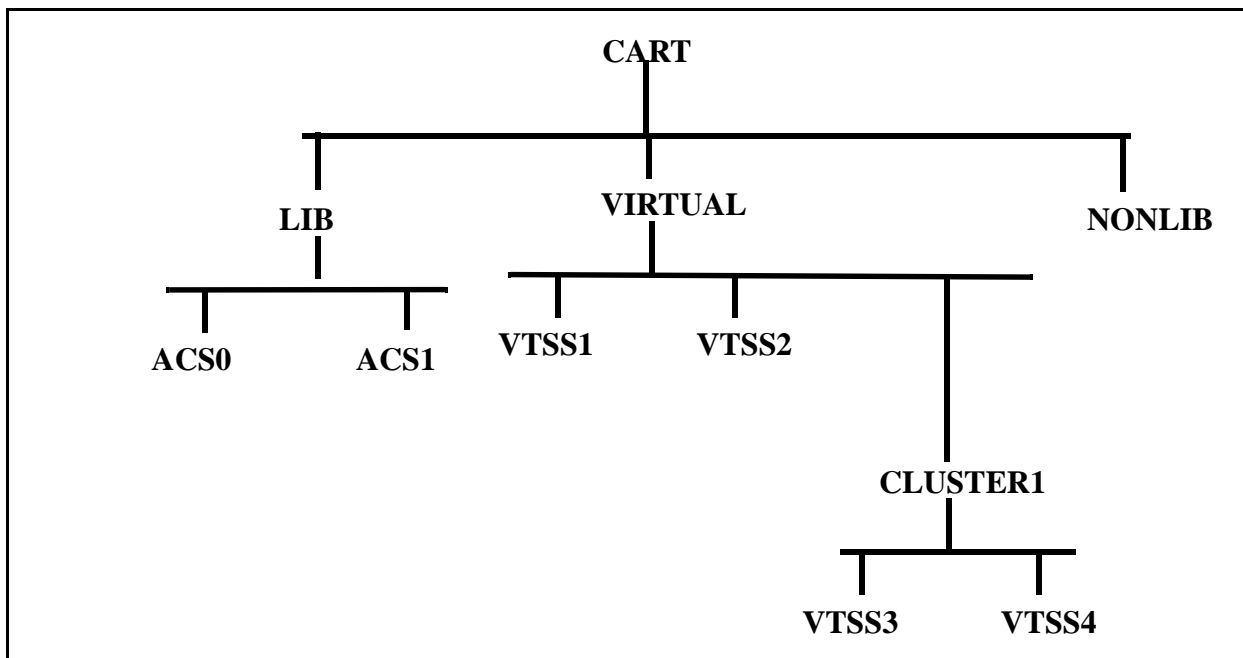


Figure 1. Example VSM Esoteric Structure for JES3 with Tape SETUP

Defining VSM Esoterics for JES3 with Tape SETUP. Figure 1 illustrates the VSM requirements and recommendations for defining esoterics in JES3 with tape SETUP as follows:

Requirement

Esoteric definition is **required** with these interfaces in this environment. That is, even if you do **not** use esoteric substitution for VSM, you must define VSM esoterics as described in the following sections.

Recommendation

StorageTek recommends that you begin the esoteric structure with a high-level esoteric that allocates all tape jobs. Your system typically has this esoteric previously defined, which is **CART** in Figure 1.

Requirement

For Nearline systems, under the high-level esoteric, you must have defined:

- A subesoteric for “library” (Nearline managed) real tape jobs (**LIB** in Figure 1). Under **LIB** are the subesoterics **ACS0** and **ACS1** that represent drives in two different ACSs. You specify **ACS0** and **ACS1** in the HSC LIBGEN on the ACSDRV parameter of the SLIACS macro.
- A second subesoteric for “non-library” (not Nearline managed) tape jobs (**NONLIB** in Figure 1). You specify this esoteric in the HSC LIBGEN on the NNLBDRV parameter of the SLILIBRY macro.

Requirement

Any esoteric that represents VTDs **must** contain only VTDs and must be defined in the JES3 initialization deck.

Requirement

For multi-VTSS environments, you **must** define an esoteric that represents all VTDs in your system (**VIRTUAL** in Figure 1. on page 15).

Requirement

You **must** place the esoteric that represents all VTDs in your system (**VIRTUAL** in Figure 1. on page 15) under the high-level esoteric. Placing the virtual subesoteric under the high-level esoteric is that this esoteric structure prevents JES3 HWS device allocation optimization from allocating a VTV to a non-virtual device.

Recommendation

For multi-VTSS environments, StorageTek recommends that you define an esoteric for each VTSS (**VTSS1**, **VTSS2**, **VTSS3**, and **VTSS4** in Figure 1. on page 15). Each esoteric *must* represent the entire range of devices for only that VTSS and the esoteric name *must* match the VTSS name on the VTD UNITATTR statement.



Note: You can also define other esoterics, such as an esoteric that represents all VTDs in all VTSSs or all VTDs in all VTSSs that comprise a VTSS Cluster (for example, **CLUSTER1** in Figure 1. on page 15). You can use these esoterics for esoteric substitution and JCL reference.

VTV Definitions

You define your system's VTV volser ranges to VTCS by volser ranges with the following:

- VTCS CONFIG
- HSC VOLATTR statement.

Determine your system's VTV volser ranges as follows:

- For an initial CONFIG definition, consider defining only enough VTVs for reasonable growth. This method allows for growth without rerunning CONFIG but does not reserve unnecessary space in the CDS, which can impact VTV processing performance. Note that if the CDS does not contain sufficient space to run VTCS CONFIG, you will also have to run HSC RECONFIG. For more information about sizing the CDS for VSM, see "HSC CDS DASD Space" on page 38.

If your VTV requirements expand beyond initial definition, then rerun CONFIG to define additional VTVs.

- You can only add new VTV ranges. A range can consist of a single volume. You cannot delete or modify existing ranges.
- If you are currently writing files to non-standard length tapes and will route these files to VTVs, you may need additional VTVs because VTVs emulate standard-length cartridges. This may require a change to the JCL volume count parameter.
- NCS/VTCS **does not allow** allocation of unlabeled tapes to VTVs. Unlabeled VTVs can cause the following for scratch VTV allocation requests:
 - If your JCL specifies a virtual esoteric, the NCS Storage Management Component (SMC) fails the allocation.
 - If you have a default esoteric such as CART and specify allocation to virtual (via TAPEREQ or HSC User Exit), the allocation will go to a non-virtual device.
- You must define VTV volsers to your tape management system; for more information, see "Updating the Tape Management System" on page 92.
- Ensure that VTV volser ranges do not duplicate or overlap existing TMS ranges or volsers of real tape volumes, including Nearline volumes, *including* MVCs and Nearline volumes that are regularly entered and ejected from the ACS!

Note: If VTDs are being used across multiple MVS images and VTV volsers are unique, add a generic entry for SYSZVOLS to the SYSTEM inclusion RNL to insure that a VTV is used by only one job at a time. If you are using automatic tape switching, also add a generic entry for SYSZVOLS to the SYSTEM inclusion RNL to prevent a system from holding a tape

device while it waits for a mount of a volume that is being used by another system.

- For more information, see the IBM publication *OS/390 MVS Planning: Global Resource Serialization*.
- If you specify scratch subpools for scratch mounts of VTVs (for example, with the TAPEREQ SUBPOOL parameter or User Exit 01), use the following guidelines:
 - If you need to define new subpools, add SCRPOOL statements to HSC PARMLIB for the VTV volsters.
 - HSC mixed-media support lets you mix VTV and real volume types in the same scratch pool. In this case, ensure that the mount request specifies a VTD as transport type (for example, via TAPEREQ MEDIA (VIRTUAL). In addition, if you are routing data to a specific VTSS (for example, by using esoteric substitution as described in “VSM Esoterics and Esoteric Substitution” on page 12) and the request specifies a subpool, ensure that the subpool contains scratch VTVs.



Hint: Note the following:

- You can use Query to display the available scratch count of a subpool.
- You can dynamically reload SCRPOOL statements via the SCRPEDEF command. For more information, see “Scratch Subpool Definition (SCRPEDEF) Command and Control Statement” in Chapter 3, “HSC Control Statements and HSC Start Procedure” of *HSC System Programmer’s Guide for MVS*.
- The Warn SCRatch, Display SCRatch, and Display THReshld commands are enhanced to let you manage and monitor scratch VTVs. For more information, see Chapter 2, “Commands, Control Statements, and Utilities,” in *HSC Operator’s Guide for MVS*.
- By default, VTCS assigns a Management Class to VTVs only on scratch mounts. You can, however, specify that VTCS assigns a Management Class whenever VTCS mounts a VTV (for read or write).



Caution: If you specify that VTCS assigns a Management Class whenever VTCS mounts a VTV, these attributes can change, which can cause undesirable or unpredictable results.

For example, if an application writes data set PROD.DATA to VTV100 with a Management Class of PROD, then writes data set TEST.DATA to VTV100 with a Management Class of TEST, then the VTV (and both data sets) has a Management Class of TEST. Similarly, it is possible to write TAPEREQ statements or SMS routines that assign different Management Classes to the same data set (for example, based on jobname), which can also cause a VTV’s Management Class to change.

RTD Unit Addresses and Identifiers

RTDs, which are Nearline transports, require UNITATTR statements and LIBGEN definitions.

If your system's RTDs are new transports, determine 4-digit hexadecimal MVS unit addresses for these transports. The addresses you choose must be the same for all hosts in the configuration. You will use these addresses to:

- Add a SLIDRIVS macro to define RTD device addresses during the HSC LIBGEN update as described in “Updating the HSC LIBGEN” on page 68.
- Create UNITATTR statements to define these transports as described in “Creating RTD UNITATTR Statements” on page 73.
- Run the HCD facility to assign MVS device numbers to these transports as described in “Assigning MVS Device Numbers to RTDs” on page 54.



Caution: Note the following:

- StorageTek **strongly recommends** that you define your RTDs to MVS (as normal 3490 tape drives), even if you do not intend to vary them online to MVS. This prevents the RTD addresses used in CONFIG and LIBGEN from accidentally being used for other devices. If you do not do this, and subsequently use the addresses for other MVS devices, you will cause problems with LOGREC processing, because VTCS will write records using the RTD addresses, and MVS will write records for other devices with those same addresses.
- StorageTek **requires** that RMM users define their RTDs to MVS. RMM causes problems if it sees the IEC501 mount message generated for RTDs by VTCS and if the device in the mount message is not defined to MVS.

Whether your system's RTDs are new or existing transports, you will use their MVS unit addresses to define the RTDs to VTCS on the CONFIG VTSS RTD DEVNO parameter.

You also specify the RTD identifier on the CONFIG VTSS RTD NAME parameter. To help identify the RTDs connected to each VTSS, StorageTek recommends that you choose RTD identifiers that reflect the VTSS identifier (specified on the VTSS NAME parameter) and the RTD's MVS device number (specified on the RTD DEVNO parameter).

In configurations where multiple VTSSs are connected to and dynamically share the same RTD, in each VTSS definition you can either assign unique RTD identifiers or use the same RTD identifier.



Note: You can specify that Nearline transports can only be used as RTDs. For more information, see “Updating the HSC LIBGEN” on page 68.

MVC Definitions

You define MVCs as described in the following sections:

- “Special VTCS Considerations for T9940B Media”
- “Special VTCS Considerations for T9840C Media” on page 21
- “Define and Select Nearline Volumes” on page 22
- “Define Available MVCs with CONFIG” on page 23
- “Define the MVC Pool” on page 24
- “Protect MVCs and Nearline Volumes” on page 25

Special VTCS Considerations for T9940B Media

T9940A and T9940B transports use the **same physical form factor** but **different recording techniques** as follows:

- T9940Bs can read from media written to by T9940As, but cannot write to T9940A media **unless** the entire volume is rewritten from beginning of tape.
- T9940As cannot read from or write to media written to by T9940Bs.

To ensure media and transport compatibility, you **must use** separate VOLATTR statements to segregate T9940A and T9940B media as follows:

- **If you have existing** T9940A media, these volumes are **already defined** with the VOLATTR MEDIA parameter value of STK2. If you are adding T9940B media, you **must** change your existing T9940A VOLATTR statements to specify MEDIA(STK2P) and RECTECH(STK2PA). You **also** need to define your new T9940B media with VOLATTR statements that specify MEDIA(STK2P) and RECTECH(STK2PB).

For example, to define MVCs MVC300-MVC599 as T9940B volumes and to redefine MVCs MVC000-MVC099 as T9940A volumes, you need the following VOLATTR statements:

```
VOLATTR SERIAL(MVC300-MVC599) MEDIA(STK2P) RECTECH(STK2PB)
VOLATTR SERIAL(MVC000-MVC099) MEDIA(STK2P) RECTECH(STK2PA)
```

- **If you are adding both** T9940A and T9940B media, you must create separate VOLATTR statements to segregate the media as follows:
 - Define the T9940A volumes with VOLATTR statements that specify MEDIA(STK2P) **and** RECTECH(STK2PA).
 - Define the T9940B volumes with VOLATTR statements that specify MEDIA(STK2P) **and** RECTECH(STK2PB).

For example, to define MVCs MVC100-MVC299 as T9940A volumes and MVCs MVC300-MVC599 as T9940B volumes, create the following VOLATTR statements:

```
VOLATTR SERIAL(MVC100-MVC299) MEDIA(STK2P) RECTECH(STK2PA)
VOLATTR SERIAL(MVC300-MVC599) MEDIA(STK2P) RECTECH(STK2PB)
```


Special VTCS Considerations for T9840C Media

T9840A/T9840B and T9840C transports use the **same physical form factor** but **different recording techniques** resulting in the following restrictions:

- T9840Cs can read from media written to by T9840As/T9840Bs, but cannot write to T9840A/T9840B media **unless** the entire volume is rewritten from beginning of tape.
- T9840As and T9840Bs cannot read from or write to media written to by T9840Cs.

To ensure media and transport compatibility, you **must use** separate VOLATTR statements to segregate T9840A/T9840B and T9840C media resulting in the following restrictions:

- **If you have existing** T9840A/T9840B media, these volumes are **already defined** with the VOLATTR MEDIA parameter value of STK1 or STK1R. If you are adding T9840C media, you **must** change your existing T9840A/T9840B VOLATTR statements to specify MEDIA(STK1R) and RECTECH(STK1RAB). You **also** need to define your new T9840C media with VOLATTR statements that specify MEDIA(STK1R) and RECTECH(STK1RC).

For example, to define MVCs MVC900-MVC999 as T9840C volumes and to redefine MVCs MVC600-MVC899 as T9840A/T9840B volumes, you need the following VOLATTR statements:

```
VOLATTR SERIAL(MVC900-MVC999) MEDIA(STK1R) RECTECH(STK1RC)
VOLATTR SERIAL(MVC600-MVC899) MEDIA(STK1R) RECTECH(STK1RAB)
```

- **If you are adding both** T9840A/T9840B media and T9840C, you must create separate VOLATTR statements to segregate the media as follows:
 - Define the T9840A/T9840B volumes with VOLATTR statements that specify MEDIA(STK1R)**and** RECTECH(STK1RAB).
 - Define the T9840C volumes with VOLATTR statements that specify MEDIA(STK1R)**and** RECTECH(STK1RC).

For example, to define MVCs MVC600-MVC899 as T9840A/T9840B volumes and MVCs MVC900-MVC999 as T9840C volumes, create the following VOLATTR statements:

```
VOLATTR SERIAL(MVC600-MVC899) MEDIA(STK1R) RECTECH(STK1RAB)
VOLATTR SERIAL(MVC900-MVC999) MEDIA(STK1R) RECTECH(STK1RC)
```

Define and Select Nearline Volumes

First, to define and select Nearline volumes for MVCs, use these guidelines:

- MVCs require VOLATTR statements to ensure that VTCS will select the correct RTD device type for each MVC. Select volumes for MVCs that are compatible with your system's RTD transport types.
- For mixed-media VSM systems, select volumes that include at least one media type compatible with each of your system's RTD transport types. See Table 9. on page 7 for information about the RTD transport types and media that VSM supports.

Note that VSM selects media for migration processing and reclaim processing according to the media types of volumes in your system's MVC pool.

- If you define new Nearline volumes as MVCs, you must create MVS volsers for these volumes and initialize STANDARD, ECART, and ZCART volumes as 36-track format standard label volumes.
- As described in "Protect MVCs and Nearline Volumes" on page 25, if possible, create a new and separate volser range for MVCs. Ensure that if you define new volumes, you do not overlap existing TMS ranges.

Define Available MVCs with CONFIG

Second, use VTCS CONFIG to define all MVCs *available* to VTCS. CONFIG reserves space for these volumes in the HSC CDS. The MVCPool statements define the *MVC pool*, which contains the MVCs that VTCS actually *uses*.

For an initial CONFIG definition, consider defining only enough MVCs for reasonable growth of your MVC pool. This method allows you to expand your MVC pool without rerunning CONFIG (you only have to change your MVCPool statements) but does not reserve unnecessary space in the CDS, which can impact MVC processing performance. Note that if the CDS does not contain sufficient space to run VTCS CONFIG, you will also have to run HSC RECONFIG. For more information about sizing the CDS for VSM, see “HSC CDS DASD Space” on page 38.

For example, if you currently need 300 MVCs but will need to add 150 more MVCs within the next 6 months, define an MVC range of 450 volsers with CONFIG, but only apply MVCPool statements to the first 300 “in use” MVCs. As your MVC space requirements increase, update and reapply your MVCPool statements to add the second 150 MVCs.

If your MVC space requirements expand beyond the second 150 MVCs, then rerun CONFIG to define additional MVC ranges and update and reapply your MVCPool statements.



Note:

- You can only add new MVC ranges. A range can consist of a single volume. You cannot delete or modify existing ranges.
- A VSM audit of all MVCs will audit all MVCs defined with CONFIG including those that are *not* specified in the MVCPool statements.

Define the MVC Pool

Third, create MVCPool statements, which specify the pool of MVCs available for migration and consolidation requests, using the following guidelines:

- Because MVCPool statements specify the “in use” MVCs, MVCPool statements can (and typically do) define a subset of the available MVCs you defined via CONFIG. MVCPool statements, however, can only specify MVCs you already defined with CONFIG. For more information about defining an initial MVC pool, see “Creating an MVC Pool” on page 73.
- StorageTek recommends that you use identical MVCPool statements on all hosts. A host can automigrate any VTV on any VTSS to which the host is connected, including VTVs created by another host. If your VSM configuration consists of hosts cross-connected to multiple VTSSs, therefore, separate MVC pools do not guarantee that a host automigrates only VTVs it creates to only its MVC pool. To most effectively segregate VTVs on groups of MVCs, see “Using Storage and Management Classes to Segregate Individual Workloads on Separate Sets of MVCs” on page 77.
- Ensure that your MVC pool consists of volumes that physically reside in ACS that contains your system’s RTDs.



Caution: In a VSM configuration with multiple hosts that share the same HSC CDS, StorageTek strongly recommends that you do *not* use HSC/VM to enter MVCs into an ACS, otherwise these MVCs will be eligible for selection as scratch volumes by any host in the configuration with HSC installed.

- To redefine your MVC pool, change your MVCPool statements and reload them via the VT MVCDEF command.

Protect MVCs and Nearline Volumes

Fourth, protect MVCs and Nearline volumes that are *not* MVCs from accidental overwrites as follows:

- If possible, create a new and separate volser range for MVCs to prevent HSC from writing to MVCs and to prevent VSM from writing to conventional Nearline volumes.
- VTCS, not MVS, controls access to MVCs. The tape management system does not control VSM access to an MVC volume and does not record its usage. If you choose to define MVCs to the tape management system, to ensure that the tape management system does not accidentally access MVCs, follow the guidelines in “Updating the Tape Management System” on page 92.
- Use your security system to restrict access to MVCs as described in “Defining MVC Pool Volser Authority” on page 90.
- HSC automatically marks newly entered MVC volumes as non-scratch. If you define existing Nearline volumes as MVCs, ensure that these volumes do not contain data you need, then run the HSC UNSCratch Utility to unscratch them. For more information, see Chapter 5. “Utility Functions” of the *HSC System Programmer’s Guide for MVS*.

MVS/CSC and Non-MVS/CSC Client Connection Values

“Virtual ACS IDs” describes the connection values you determine for MVS/CSC and non-MVS/CSC clients.

Virtual ACS IDs

To connect to VSM, clients use a decimal virtual ACS ID that maps to a VTSS identifier that you determined in “VTSS Identifiers” on page 10. You use the LibraryStation VIRTACS statement to do this mapping. To avoid conflicts with real ACS IDs, StorageTek recommends that you select virtual ACS IDs by starting with the highest possible value (126) and working backwards.

For example, to define virtual ACS 126 and map it to VTSS VTSS02, you would create the following VIRTACS statement:

```
VIRTACS ID(126) VTSSNAME(VTSS02)
```

Virtual ACS Location for VSM2s and VSM3s. To LibraryStation, a VSM2/3 virtual ACS consists of four virtual LSMs (0-3), each containing four drive panels (1-4) with four VTDs each (0-3) for a total of 64 VTDs. LibraryStation, therefore, references VTDs by their ACS location in *acsid,lsmid,panelnum,devicenum* format. For example, LibraryStation references the 27th VTD in virtual ACS 126, above, by ACS location 126,1,3,2.

Table 11 cross-references VTD numbers to their LibraryStation virtual ACS locations. See “Connecting MVS/CSC Clients to VSM” on page 84 for procedures for defining these library locations to MVS/CSC clients and mapping them to MVS device numbers.

Table 11. LibraryStation VSM2/3 Virtual ACS Locations for VTDs

VTD	Virtual ACS Location
1	<i>acsid,0,1,0</i>
2	<i>acsid,0,1,1</i>
3	<i>acsid,0,1,2</i>
4	<i>acsid,0,1,3</i>
5	<i>acsid,0,2,0</i>
6	<i>acsid,0,2,1</i>
7	<i>acsid,0,2,2</i>
8	<i>acsid,0,2,3</i>
9	<i>acsid,0,3,0</i>
10	<i>acsid,0,3,1</i>
11	<i>acsid,0,3,2</i>
12	<i>acsid,0,3,3</i>
13	<i>acsid,0,4,0</i>

14	<i>acsid</i> ,0,4,1
15	<i>acsid</i> ,0,4,2
16	<i>acsid</i> ,0,4,3
17	<i>acsid</i> ,1,1,0
18	<i>acsid</i> ,1,1,1
19	<i>acsid</i> ,1,1,2
20	<i>acsid</i> ,1,1,3
21	<i>acsid</i> ,1,2,0
22	<i>acsid</i> ,1,2,1
23	<i>acsid</i> ,1,2,2
24	<i>acsid</i> ,1,2,3
25	<i>acsid</i> ,1,3,0
26	<i>acsid</i> ,1,3,1
27	<i>acsid</i> ,1,3,2
28	<i>acsid</i> ,1,3,3
29	<i>acsid</i> ,1,4,0
30	<i>acsid</i> ,1,4,1
31	<i>acsid</i> ,1,4,2
32	<i>acsid</i> ,1,4,3
33	<i>acsid</i> ,2,1,0
34	<i>acsid</i> ,2,1,1
35	<i>acsid</i> ,2,1,2
36	<i>acsid</i> ,2,1,3
37	<i>acsid</i> ,2,2,0
38	<i>acsid</i> ,2,2,1
39	<i>acsid</i> ,2,2,2
40	<i>acsid</i> ,2,2,3
41	<i>acsid</i> ,2,3,0
42	<i>acsid</i> ,2,3,1
43	<i>acsid</i> ,2,3,2
44	<i>acsid</i> ,2,3,3
45	<i>acsid</i> ,2,4,0

46	<i>acsid,2,4,1</i>
47	<i>acsid,2,4,2</i>
48	<i>acsid,2,4,3</i>
49	<i>acsid,3,1,0</i>
50	<i>acsid,3,1,1</i>
51	<i>acsid,3,1,2</i>
52	<i>acsid,3,1,3</i>
53	<i>acsid,3,2,0</i>
54	<i>acsid,3,2,1</i>
55	<i>acsid,3,2,2</i>
56	<i>acsid,3,2,3</i>
57	<i>acsid,3,3,0</i>
58	<i>acsid,3,3,1</i>
59	<i>acsid,3,3,2</i>
60	<i>acsid,3,3,3</i>
61	<i>acsid,3,4,0</i>
62	<i>acsid,3,4,1</i>
63	<i>acsid,3,4,2</i>
64	<i>acsid,3,4,3</i>

Virtual ACS Location for VSM4s. To LibraryStation, a VSM4 virtual ACS consists of sixteen virtual LSMs (0-15), each containing four drive panels (1-4) with four VTDs each (0-3) for a total of 64 VTDs. LibraryStation, therefore, references VTDs by their ACS location in *acsid, lsmid, panelnum, devicenum* format. For example, LibraryStation references the 27th VTD in virtual ACS 126, above, by ACS location 126,1,3,2.

Table 12 cross-references VTD numbers to their LibraryStation virtual ACS locations. See “Connecting MVS/CSC Clients to VSM” on page 91 for procedures for defining these library locations to MVS/CSC clients and mapping them to MVS device numbers.

Table 12. LibraryStation VSM4 Virtual ACS Locations for VTDs

VTD	Virtual ACS Location
1	<i>acsid,0,1,0</i>
2	<i>acsid,0,1,1</i>
3	<i>acsid,0,1,2</i>
4	<i>acsid,0,1,3</i>
5	<i>acsid,0,2,0</i>
6	<i>acsid,0,2,1</i>
7	<i>acsid,0,2,2</i>
8	<i>acsid,0,2,3</i>
9	<i>acsid,0,3,0</i>
10	<i>acsid,0,3,1</i>
11	<i>acsid,0,3,2</i>
12	<i>acsid,0,3,3</i>
13	<i>acsid,0,4,0</i>
14	<i>acsid,0,4,1</i>
15	<i>acsid,0,4,2</i>
16	<i>acsid,0,4,3</i>
17	<i>acsid,1,1,0</i>
18	<i>acsid,1,1,1</i>
19	<i>acsid,1,1,2</i>
20	<i>acsid,1,1,3</i>
21	<i>acsid,1,2,0</i>
22	<i>acsid,1,2,1</i>

23	<i>acsid,1,2,2</i>
24	<i>acsid,1,2,3</i>
25	<i>acsid,1,3,0</i>
26	<i>acsid,1,3,1</i>
27	<i>acsid,1,3,2</i>
28	<i>acsid,1,3,3</i>
29	<i>acsid,1,4,0</i>
30	<i>acsid,1,4,1</i>
31	<i>acsid,1,4,2</i>
32	<i>acsid,1,4,3</i>
33	<i>acsid,2,1,0</i>
34	<i>acsid,2,1,1</i>
35	<i>acsid,2,1,2</i>
36	<i>acsid,2,1,3</i>
37	<i>acsid,2,2,0</i>
38	<i>acsid,2,2,1</i>
39	<i>acsid,2,2,2</i>
40	<i>acsid,2,2,3</i>
41	<i>acsid,2,3,0</i>
42	<i>acsid,2,3,1</i>
43	<i>acsid,2,3,2</i>
44	<i>acsid,2,3,3</i>
45	<i>acsid,2,4,0</i>
46	<i>acsid,2,4,1</i>
47	<i>acsid,2,4,2</i>
48	<i>acsid,2,4,3</i>
49	<i>acsid,3,1,0</i>
50	<i>acsid,3,1,1</i>
51	<i>acsid,3,1,2</i>
52	<i>acsid,3,1,3</i>
53	<i>acsid,3,2,0</i>
54	<i>acsid,3,2,1</i>

55	<i>acsid,3,2,2</i>
56	<i>acsid,3,2,3</i>
57	<i>acsid,3,3,0</i>
58	<i>acsid,3,3,1</i>
59	<i>acsid,3,3,2</i>
60	<i>acsid,3,3,3</i>
61	<i>acsid,3,4,0</i>
62	<i>acsid,3,4,1</i>
63	<i>acsid,3,4,2</i>
64	<i>acsid,3,4,3</i>
65	<i>acsid,4,1,0</i>
66	<i>acsid,4,1,1</i>
67	<i>acsid,4,1,2</i>
68	<i>acsid,4,1,3</i>
69	<i>acsid,4,2,0</i>
70	<i>acsid,4,2,1</i>
71	<i>acsid,4,2,2</i>
72	<i>acsid,4,2,3</i>
73	<i>acsid,4,3,0</i>
74	<i>acsid,4,3,1</i>
75	<i>acsid,4,3,2</i>
76	<i>acsid,4,3,3</i>
77	<i>acsid,4,4,0</i>
78	<i>acsid,4,4,1</i>
79	<i>acsid,4,4,2</i>
80	<i>acsid,4,4,3</i>
81	<i>acsid,5,1,0</i>
82	<i>acsid,5,1,1</i>
83	<i>acsid,5,1,2</i>
84	<i>acsid,5,1,3</i>
85	<i>acsid,5,2,0</i>
86	<i>acsid,5,2,1</i>

87	<i>acsid,5,2,2</i>
88	<i>acsid,5,2,3</i>
89	<i>acsid,5,3,0</i>
90	<i>acsid,5,3,1</i>
91	<i>acsid,5,3,2</i>
92	<i>acsid,5,3,3</i>
93	<i>acsid,5,4,0</i>
94	<i>acsid,5,4,1</i>
95	<i>acsid,5,4,2</i>
96	<i>acsid,5,4,3</i>
97	<i>acsid,6,1,0</i>
98	<i>acsid,6,1,1</i>
99	<i>acsid,6,1,2</i>
100	<i>acsid,6,1,3</i>
101	<i>acsid,6,2,0</i>
102	<i>acsid,6,2,1</i>
103	<i>acsid,6,2,2</i>
104	<i>acsid,6,2,3</i>
105	<i>acsid,6,3,0</i>
106	<i>acsid,6,3,1</i>
107	<i>acsid,6,3,2</i>
108	<i>acsid,6,3,3</i>
109	<i>acsid,6,4,0</i>
110	<i>acsid,6,4,1</i>
111	<i>acsid,6,4,2</i>
112	<i>acsid,6,4,3</i>
113	<i>acsid,7,1,0</i>
114	<i>acsid,7,1,1</i>
115	<i>acsid,7,1,2</i>
116	<i>acsid,7,1,3</i>
117	<i>acsid,7,2,0</i>
118	<i>acsid,7,2,1</i>

119	<i>acsid,7,2,2</i>
120	<i>acsid,7,2,3</i>
121	<i>acsid,7,3,0</i>
122	<i>acsid,7,3,1</i>
123	<i>acsid,7,3,2</i>
124	<i>acsid,7,3,3</i>
125	<i>acsid,7,4,0</i>
126	<i>acsid,7,4,1</i>
127	<i>acsid,7,4,2</i>
128	<i>acsid,7,4,3</i>
129	<i>acsid,8,1,0</i>
130	<i>acsid,8,1,1</i>
131	<i>acsid,8,1,2</i>
132	<i>acsid,8,1,3</i>
133	<i>acsid,8,2,0</i>
134	<i>acsid,8,2,1</i>
135	<i>acsid,8,2,2</i>
136	<i>acsid,8,2,3</i>
137	<i>acsid,8,3,0</i>
138	<i>acsid,8,3,1</i>
139	<i>acsid,8,3,2</i>
140	<i>acsid,8,3,3</i>
141	<i>acsid,8,4,0</i>
142	<i>acsid,8,4,1</i>
143	<i>acsid,8,4,2</i>
144	<i>acsid,8,4,3</i>
145	<i>acsid,9,1,0</i>
146	<i>acsid,9,1,1</i>
147	<i>acsid,9,1,2</i>
148	<i>acsid,9,1,3</i>
149	<i>acsid,9,2,0</i>
150	<i>acsid,9,2,1</i>

151	<i>acsid,9,2,2</i>
152	<i>acsid,9,2,3</i>
153	<i>acsid,9,3,0</i>
154	<i>acsid,9,3,1</i>
155	<i>acsid,9,3,2</i>
156	<i>acsid,9,3,3</i>
157	<i>acsid,9,4,0</i>
158	<i>acsid,9,4,1</i>
159	<i>acsid,9,4,2</i>
160	<i>acsid,9,4,3</i>
161	<i>acsid,10,1,0</i>
162	<i>acsid,10,1,1</i>
163	<i>acsid,10,1,2</i>
164	<i>acsid,10,1,3</i>
165	<i>acsid,10,2,0</i>
166	<i>acsid,10,2,1</i>
167	<i>acsid,10,2,2</i>
168	<i>acsid,10,2,3</i>
169	<i>acsid,10,3,0</i>
170	<i>acsid,10,3,1</i>
171	<i>acsid,10,3,2</i>
172	<i>acsid,10,3,3</i>
173	<i>acsid,10,4,0</i>
174	<i>acsid,10,4,1</i>
175	<i>acsid,10,4,2</i>
176	<i>acsid,10,4,3</i>
177	<i>acsid,11,1,0</i>
178	<i>acsid,11,1,1</i>
179	<i>acsid,11,1,2</i>
180	<i>acsid,11,1,3</i>
181	<i>acsid,11,2,0</i>
182	<i>acsid,11,2,1</i>

183	<i>acsid,11,2,2</i>
184	<i>acsid,11,2,3</i>
185	<i>acsid,11,3,0</i>
186	<i>acsid,11,3,1</i>
187	<i>acsid,11,3,2</i>
188	<i>acsid,11,3,3</i>
189	<i>acsid,11,4,0</i>
190	<i>acsid,11,4,1</i>
191	<i>acsid,11,4,2</i>
192	<i>acsid,11,4,3</i>
193	<i>acsid,12,1,0</i>
194	<i>acsid,12,1,1</i>
195	<i>acsid,12,1,2</i>
196	<i>acsid,12,1,3</i>
197	<i>acsid,12,2,0</i>
198	<i>acsid,12,2,1</i>
199	<i>acsid,12,2,2</i>
200	<i>acsid,12,2,3</i>
201	<i>acsid,12,3,0</i>
202	<i>acsid,12,3,1</i>
203	<i>acsid,12,3,2</i>
204	<i>acsid,12,3,3</i>
205	<i>acsid,12,4,0</i>
206	<i>acsid,12,4,1</i>
207	<i>acsid,12,4,2</i>
208	<i>acsid,12,4,3</i>
209	<i>acsid,13,1,0</i>
210	<i>acsid,13,1,1</i>
211	<i>acsid,13,1,2</i>
212	<i>acsid,13,1,3</i>
213	<i>acsid,13,2,0</i>
214	<i>acsid,13,2,1</i>

215	<i>acsid,13,2,2</i>
216	<i>acsid,13,2,3</i>
217	<i>acsid,13,3,0</i>
218	<i>acsid,13,3,1</i>
219	<i>acsid,13,3,2</i>
220	<i>acsid,13,3,3</i>
221	<i>acsid,13,4,0</i>
222	<i>acsid,13,4,1</i>
223	<i>acsid,13,4,2</i>
224	<i>acsid,13,4,3</i>
225	<i>acsid,14,1,0</i>
226	<i>acsid,14,1,1</i>
227	<i>acsid,14,1,2</i>
228	<i>acsid,14,1,3</i>
229	<i>acsid,14,2,0</i>
230	<i>acsid,14,2,1</i>
231	<i>acsid,14,2,2</i>
232	<i>acsid,14,2,3</i>
233	<i>acsid,14,3,0</i>
234	<i>acsid,14,3,1</i>
235	<i>acsid,14,3,2</i>
236	<i>acsid,14,3,3</i>
237	<i>acsid,14,4,0</i>
238	<i>acsid,14,4,1</i>
239	<i>acsid,14,4,2</i>
240	<i>acsid,14,4,3</i>
241	<i>acsid,15,1,0</i>
242	<i>acsid,15,1,1</i>
243	<i>acsid,15,1,2</i>
244	<i>acsid,15,1,3</i>
245	<i>acsid,15,2,0</i>
246	<i>acsid,15,2,1</i>

247	<i>acsid,15,2,2</i>
248	<i>acsid,15,2,3</i>
249	<i>acsid,15,3,0</i>
250	<i>acsid,15,3,1</i>
251	<i>acsid,15,3,2</i>
252	<i>acsid,15,3,3</i>
253	<i>acsid,15,4,0</i>
254	<i>acsid,15,4,1</i>
255	<i>acsid,15,4,2</i>
256	<i>acsid,15,4,3</i>

HSC CDS DASD Space

Before installing VTCS, you must calculate the DASD space required for the HSC control data set (CDS). The DASD space for the CDS must be increased to accommodate your VSM system's resource definitions. The additional number of 4k blocks required in the CDS for VTCS can be expressed as:

- **For HSC 4.0 format CDSs:**

$(\text{number of VTVs} / 58) + (\text{number of MVCs} / 71) + 17(\text{number of VTSS}) + \text{number of configured MVC ranges} + \text{number of configured VTV ranges} + 13$

- **For VSM Extended Format CDSs (HSC 5.0 and above):**

$(\text{number of VTVs} / 23) + (\text{number of MVCs} / 37) + 17(\text{number of VTSS}) + \text{number of configured MVC ranges} + \text{number of configured VTV ranges} + 13$



Hint: Contact StorageTek Software Support for a tool that calculates DASD space for CDSs using the algorithms shown above.

Tape Management System DASD Space

To accommodate your VSM system's VTVs, you may need to increase the DASD space for your tape management system. After you determine the number and range of VTVs your VSM system requires, see your tape management system documentation for specific information on calculating the DASD space requirements.

VSM Candidate Data Sets

Your StorageTek representative will run the VSM pre-sales planning tool to identify VSM candidate data sets. You choose a method to route these data sets to VSM as described in "Routing Data Sets to VSM" on page 93.

HSC COMMPath METHod Value

To optimize performance, StorageTek recommends that you set the HSC COMMPath METHod parameter to either LMU or VTAM, *not* to CDS to allow even sharing of resources in a multi-host configuration as shown in the example in Figure 22 on page 82.

Data Chaining a VTD Read Forward or Write Command

Note that when data chaining a Read Forward or Write command, the VTSS requires the minimum data chained update count.

VSM Policies

The following sections describe VSM operating policies:

- “VTSS Management Policies” on page 39
- “VTV Migration and Consolidation Policies” on page 43
- “VTV Recall Policy” on page 48
- “MVC Policies” on page 49

VTSS Management Policies

The following sections describe these VTSS management policies:

- “Maximum and Minimum Concurrent Automatic Migration, Immediate Migration, and Migrate-to-Threshold Tasks”
- “AMT Settings” on page 40
- “Deleting Scratched VTVs” on page 42

Maximum and Minimum Concurrent Automatic Migration, Immediate Migration, and Migrate-to-Threshold Tasks. These configuration parameters specify the maximum and minimum number of concurrent automatic migration, immediate migration, and migrate-to-threshold tasks for each VTSS. Use these parameters to balance migration tasks with other tasks (such as recall and reclaim) for the RTDs you have defined for each VTSS. To set these parameters:

- Use the `CONFIG VTSS MAXMIG` parameter to specify the maximum migration tasks. You can also use `SET MIGOPT` to change this value.
- Use the `CONFIG VTSS MINMIG` parameter to specify the minimum migration tasks. You can also use `SET MIGOPT` to change this value.



Note: In some situations, VTCS may not be able to activate *all* the migration tasks specified by the `MAXMIG` parameter. For example:

- The VSM-wide RTD configuration consists of four 9840 and four 9490 transports.
- No Storage Classes with `STK1R` as the primary media have been defined.
- There is sufficient MVC media for the 9490 transports.

In this configuration, because only the 9490 media is used, only a maximum of four migration tasks are activated using the 9490 RTDs.

Similarly, there are circumstances when VTCS will start *less* than the number of migration tasks specified by the MINMIG parameter. For example:

- The configuration consists of a single VTSS with 4 RTDs in ACS 0 and 4 RTDs in ACS 1. All RTD device types are identical.
- MINMIG and MAXMIG are both set to 8.
- Two Storage Classes are defined, which point, respectively, to ACS 0 and ACS 1.

In this configuration, if there are migrates queued for both Storage Classes, then VTCS will start 8 requests. If however, there are only migrations queued for one Storage Class, then VTCS *will not* start 8 requests because the workload can only be serviced by one Storage Class and this class can only run on 4 RTDs.

Finally, also note that when you reset the MINMIG and/or MAXMIG values, the actual number of migration tasks may not be immediately affected because of the way that VTCS manages migration tasks.

You can use Query to display migration status.



AMT Settings. This policy controls the automatic space management/migration cycle. This cycle begins when DBU (Disk Buffer Utilization) exceeds the high AMT (HAMT) or the number of VTVs exceeds 97,000 and continues until DBU drops below the low AMT (LAMT).

To set the AMTs, use either of the following:

- The CONFIG VTSS LOW and HIGH parameters.
- SET MIGOPT.



Note:

- With CONFIG, AMT settings take effect when you start HSC and apply to the specified VTSS.

- With SET MIGOPT:
 - AMT settings take effect immediately and apply to the specified VTSS or if no VTSS is specified, to all VTSSs. If you try to set global values (no VTSS specified) and the values are not valid for one VTSS (for example, MAXMIG(5) and one VTSS only has 4 RTDs connected), VTCS will not set values for any VTSSs
 - You can set the LAMT, the HAMT, or both.

The default high and low AMTs are 80% and 70% respectively. Valid values are 5 to 95 for the LAMT and 6 to 95 for the HAMT. The LAMT must be at least one less than the HAMT. For example, for a HAMT of 90%, you cannot specify a LAMT greater than 89%.

The following are general guidelines for changing the defaults:

- The *difference* between the high and low AMTs affects the duration of the space management/migration cycle.
- *Lowering* the HAMT tends to trigger *more frequent* space management/migration cycles.
- *Raising* the HAMT tends to trigger *less frequent* space management/migration cycles.
- *Lowering* the LAMT tends to free more VTSS space *and* migrate *more* VTVs.
- *Raising* the LAMT tends to keep more VTVs resident in VTSS space *and* migrate *fewer* VTVs.



Hint: You can use Query to display the DBU, HAMT, and LAMT for each VTSS in your system. You can also use Query to display migration status.

Deleting Scratched VTVs. Use the DELSCR parameter of the MGMTclas statement to specify whether VSM deletes scratched VTVs.

Specifying DELSCR YES causes VSM to delete scratched VTVs, which frees VTSS buffer space.



Warning: When you scratch a VTV with DELSCR YES attribute, **VSM erases the VTV data at scratch synchronization time**, which eliminates the ability to “unscratch” a VTV to recover data!

Also note that when using previous releases of HSC SLUCONDB to perform scratch synchronization, SLUCONDB attempted to scratch everything that was marked scratch in the TMS database. For HSC 5.1, however, SLUCONDB has been updated to scratch *only* those volumes that are not in scratch status in the HSC CDS. Therefore, for HSC 5.1, the *only* possibilities of inadvertently scratching a VTV resulting in data loss at scratch synchronization time are as follows:

- If you are running the HSC SLUADMIN Scratch Update Utility at the same time that SLUCONDB is running.
- If you do not specify the current TMS database and/or the current HSC CDS when using SLUCONDB.

For more information about HSC scratch synchronization with the Scratch Conversion Utility (SLUCONDB), see Chapter 5, “Utility Functions” of *HSC System Programmer’s Guide for MVS*.

Also note that for HSC and MVS/CSC, the DELDISP parameter has two values that affect how HSC manages the scratch status of VTVs and real volumes that were mounted scratch and the delete disposition on the dismount message is delete ('D').

For more information about ExLM scratch synchronization with the SYNCVTV function, see “Using ExLM with VTCS (All Versions)” in Chapter 2, “Using ExLM to Manage Nearline and VTCS Resources” of *ExLM System Administrator’s Guide*.

VTV Migration and Consolidation Policies

The following sections describe these VTV migration policies:

- “Hosts Disabled from Migration, Consolidation and Export by VTV or Management Class” on page 43
- “VTV Residency Interval before Automatic Migration Candidacy” on page 44
- “Immediately Migrate VTVs On Dismount” on page 45
- “MVC Retain Interval” on page 46
- “Maximum VTVs per MVC” on page 46
- “ACS and Media Type of MVCs for Migration and Reclamation.” on page 46
- “VTV Duplexing” on page 46
- “Migrate Duplexed VTVs to Separate ACSs” on page 47
- “Output MVC ACS and Media for VTV Consolidation” on page 47
- “Source MVC ACS and Media for Consolidation of Migrated Duplexed VTVs” on page 47



Note: You can use the VTVMaint utility to change a VTV’s Management and Storage Class.

Hosts Disabled from Migration, Consolidation and Export by VTV or Management Class. You can specify that a host cannot initiate automatic and demand migration and consolidation processing. Use the CONFIG HOST NOMIGRAT parameter to set this policy. Note that NOMIGRAT also disables a host from doing exports by VTV or Management Class.



Note: Specifying NOMIGRAT also causes NORECLAM to be set; for more information, see “Hosts Disabled for Reclamation” on page 49.

IMMEDmig KEEP and IMMEdmig DELETE are mutually exclusive with CONFIG HOST NOMIGRAT. If you specify both, the IMMEdmig value overrides NOMIGRAT (for only those VTVs with the IMMEdmig value), and VTCS does not issue a message about this override.

VTV Residency Interval before Automatic Migration Candidacy. By default, VSM selects VTVs for migration. You can, however, specify how long (in hours) VTCS attempts to keep a VTV as VTSS-resident before it becomes an automatic migration candidate.

Use the `RESTIME` parameter of the `MGMTclas` statement to set this policy. The `RESTIME` value in a VTV's Management Class sets the recommended interval that the VTV remains VTSS-resident from the time that instance of the VTV is created. A new instance of the VTV is created whenever the VTV is updated. At automigration time, the creation date and time of the VTV instance plus the `RESTIME` value is compared to the TOD clock to determine if the VTV is an automatic migration candidate.



Note: Note the following:

- A VTV's Management Class (and attributes, such as `RESTIME`) is set after a scratch mount or optionally after a specific mount if `VTVattr = ALLmount`. For more information, see page 18.
- The `RESTIME` value is *only* a recommendation. VTCS can migrate a VTV before its residency interval expires if the DBU has not reached the `LAMT` or the specified migrate-to-threshold value and no VTVs have expired their residency intervals.
- You can do a demand migrate of a VTV and delete it from the VTSS even if its residency interval has not expired.
- The `RESTIME` and `IMMEDmig (DELETE)` parameters are mutually exclusive; for more information, see “Immediately Migrate VTVs On Dismount” on page 45.

The following example shows how the `RESTIME` parameter works:

1. You create Management Class with a `RESTIME` of 10 hours.
2. A job requests a scratch mount for the Management Class you created in Step 1. VTCS selects and mounts a scratch VTV. The VTV is updated, so at dismount time, its `RESTIME` value is set to 10 hours (which began when VTCS mounted the VTV).
3. VTCS migrates the VTV after 3 hours, then recalls the VTV 2 hours later for a read. The `RESTIME` value is *not* reset, and there are now 5 hours of residency remaining.
4. 2 hours later, a job updates the VTV, which was 7 hours old. The update creates a new instance of the VTV and the residency interval will restart from the time the VTV was mounted for update.

5. 24 hours later, VTCS migrates the VTV, then recalls it 2 days later for a read. VTCS does *not* create a new instance of the VTV because it is not updated. The residency interval has expired and the VTV is therefore an automatic migration candidate based only on least-recently-used/size criteria.
6. A week later, the VTV is scratched. VTCS eventually selects and mounts the VTV to satisfy a scratch mount request. If the VTV is updated, its residency interval is set to the RESTIME value of the Management Class being used.

Immediately Migrate VTVs On Dismount. With the IMMEdmig parameter of the MGMTclas statement, you can specify whether VSM will immediately schedule a VTV for migration after dismounting it. When the migration actually occurs depends on RTD availability, Storage Classes for immediate migration, and the total number of immediate migrates scheduled.

The following are guidelines for setting IMMEdmig:

- Specify NO (the default) if you *do not* want immediately migration and you *do* want other migration policies to determine your migration strategy.
- Specify KEEP if you want immediate migration and want to keep copies of the migrated VTVs resident on the VTSS until they become eligible for deletion.



Caution: IMMEdmig KEEP ensures that VTVs are immediately migrated and kept VTSS-resident; however, it does not free up VTSS space, may increase I/O to the RTDs, uses up MVC space more quickly, and may also increase the need for MVC space reclamation.

- Specify DELETE if you want immediate migration and want to delete VTVs from the VTSS after migration.



Caution: IMMEdmig DELETE ensures that VTVs are immediately migrated and frees VTSS space; however, it preferences migration processing, may increase I/O to the RTDs, uses up MVC space more quickly, and may also increase the need for MVC space reclamation and VTV recalls.

- The RESTIME and IMMEdmig(DELETE) parameters are mutually exclusive; for more information, see “VTV Migration and Consolidation Policies” on page 43.
- IMMEdmig KEEP and IMMEdmig DELETE are mutually exclusive with CONFIG HOST NOMIGRAT. If you specify both, the IMMEdmig value overrides NOMIGRAT (for only those VTVs with the IMMEdmig value), and VTCS does not issue a message about this override.
- If VTCS stops with pending immediate migrations, these migrations will resume when VTCS restarts.

For information on setting this policy with the MGMTclas statement, see “Creating and Using VSM Management and Storage Classes” on page 74.

MVC Retain Interval. This policy specifies how long VTCS will retain an MVC on an RTD in idle mode after a migration. Retaining the MVC can reduce MVC mounts.



Note: When VTCS shuts down, VTCS dismounts all MVCs regardless of the MVC retain interval.

Use the `CONFIG VTSS RETAIN` parameter to set this policy.

Maximum VTVs per MVC. This policy specifies the maximum number of VTVs that VSM will migrate to a migration MVC or consolidate to a consolidation MVC. This policy applies to all MVCs and, at the time you set the policy, applies only to future migrations. That is, it will not lower the number of VTVs already migrated to an MVC. If the policy is not specified, the default is 32000 VTVs per MVC unless the available MVC space is less than any remaining current VTSS resident VTV.

Generally, use the default setting to allow VSM to automatically manage VTV stacking. However, specifying a maximum value lower than the default may improve recall performance in some situations (for example, in a VSM system where all MVCs are type STK2P). Note, however, that a very low maximum value can reduce that percentage of usable MVC space. If the maximum VTVs per MVC is exceeded, then usable space is reported as 0%.

Note: Use the `CONFIG GLOBAL MAXVTV` parameter to set this policy.

ACS and Media Type of MVCs for Migration and Reclamation. You can specify the ACS and a preference list of media types for MVCs for migration and reclamation with the `MIGpol` parameter of the `MGMTclas` statement.

For procedures to set this policy, see “Creating and Using VSM Management and Storage Classes” on page 74.

VTV Duplexing. This policy specifies whether VSM migrates two copies of a VTV to two separate MVCs. Duplexing provides two copies of a VTV on separate physical cartridges, but it also doubles the MVC space required for migrated VTVs. Note that a consolidated VTV will *not* be duplexed for migration, even if duplexing is specified for migration; only one copy will be migrated.

You specify duplexing for one or more VTVs on either the `DUPlex` or `MIGpol` parameters of the `MGMTclas` statement.



Note: If the Management Class you use for consolidation specifies the `DUPlex` parameter, duplexing is ignored for consolidation for this Management Class but duplexing *is* supported for migration for this Management Class.

For procedures to set this policy, see “Creating and Using VSM Management and Storage Classes” on page 74.

Migrate Duplexed VTVs to Separate ACSs. You can specify whether duplexed VTVs go to different MVCs in the same ACS or different MVCs in separate ACSs with the `ACSlist` parameter of the `MGMTclas` statement.

For procedures to set this policy, see “Creating and Using VSM Management and Storage Classes” on page 74.

Output MVC ACS and Media for VTV Consolidation.

This policy specifies the Storage Class that determines the output MVC ACS and media for VTV consolidation. You set this policy on the `CONTGT` parameter of the `MGMTclas` statement.

For procedures to set this policy, see “Creating and Using VSM Management and Storage Classes” on page 74 and “Using Storage and Management Classes to Specify the Source and Target MVC for VTV Consolidation” on page 78.

Source MVC ACS and Media for Consolidation of Migrated Duplexed VTVs.

This policy specifies the Storage Class that determines the source MVC ACS and media for consolidation of VTVs that are migrated and duplexed to two different MVC locations or media types. You set this policy on the `MIGpol` and `CONSRC` parameters of the `MGMTclas` statement.

For procedures to set this policy, see “Creating and Using VSM Management and Storage Classes” on page 74 and “Using Storage and Management Classes to Specify the Source and Target MVC for VTV Consolidation” on page 78.

VTV Recall Policy

By default, VTCS recalls VTVs with read data checks. You can, however, specify whether VTCS recalls VTVs with read data checks on:

- The GLOBAL statement of CONFIG.
- RECALL.
- MVCDRAIN.
- CONSolid.
- EXPORT.



During MVC reclaims, VTCS will never recall VTVs with read data checks, regardless of the RECALWER setting on the CONFIG GLOBAL statement.

MVC Policies

The following sections describe these policies:

- “Maximum MVCs Concurrently Processed for Reclamation and Drain”
- “Hosts Disabled for Reclamation”
- “Free MVCs Threshold - Starts Automatic Space Reclamation” on page 50
- “MVC Fragmented Space Threshold- Determines MVC Eligibility for Reclamation” on page 50
- “Eligible/Total MVCs Threshold - Starts Automatic Space Reclamation” on page 51
- “Maximum MVCs Processed Per Reclaim” on page 52



Note: Reclamation turns fragmented MVC space (space that contains non-current VTVs) into usable space (writable MVC space). MVC reports and Query show the percentages of MVC space that is fragmented, used (space that contains current VTVs), available, and usable. Note that usable space may be zero even if there is still space physically available. For example, if the maximum VTVs per MVC is exceeded, then usable space is reported as 0%. You set maximum VTVs per MVC as described in “Maximum VTVs per MVC” on page 46. Similarly, if a data check error has been reported against an MVC, VTCS will not use this MVC for output and usable space is reported as 0%.

Maximum MVCs Concurrently Processed for Reclamation and Drain. This policy specifies the maximum number of MVCs concurrently processed for reclamation and drain. Use the CONFIG RECLAIM CONMVC parameter to set this policy. Valid values for the CONMVC parameter are 1 to 99. The default is 1. You can also use the following to override the value specified on the CONFIG RECLAIM CONMVC parameter:

- MVCDRAIN
- RECLAIM

Hosts Disabled for Reclamation. You can specify that a host cannot initiate automatic or demand reclaim processing (the host can still do demand MVC drains via MVCDRAIN). Use the CONFIG HOST NORECLAM parameter to set this policy.

Free MVCs Threshold - Starts Automatic Space Reclamation. This policy specifies the minimum number of free MVCs in the MVC pool. A free MVC has 100% usable space and does not contain any migrated VTVs.

Use the CONFIG GLOBAL MCVFREE parameter to set this policy. Valid values are 0 to 255. The default is 40.

VTCS checks this value for each ACS. VTCS issues message SLS6616I and starts an automatic space reclamation if *both* of the following occurs:

- Free MVCs is equal to or less than the value specified on CONFIG MCVFREE.
- There is at least one eligible MVC as defined by the CONFIG RECLAIM THRESHLD parameter; for more information, see “MVC Fragmented Space Threshold- Determines MVC Eligibility for Reclamation” on page 50.



Note: If you set MCVFREE=0, VTCS actually uses the default value (40).



Hint: StorageTek recommends that you ensure that your MVC pool always has *at least* one eligible MVC for each MVC media type.

Otherwise, you may need to change the CONFIG GLOBAL MCVFREE value, add more MVCs to the pool, or both. You can use Query to display the number of free MVCs in your MVC pool.

MVC Fragmented Space Threshold- Determines MVC Eligibility for Reclamation. This policy specifies the fragmented space threshold (as a percentage) that determines when an MVC is eligible for demand or automatic reclamation.

Use the CONFIG RECLAIM THRESHLD parameter to set this policy. If fragmented space on an MVC exceeds the value specified on CONFIG THRESHLD, VTCS makes the MVC eligible for reclamation. You can use RECLAIM to change the CONFIG settings for the THRESHLD parameter.

Regardless of the percentage of fragmented space on an MVC versus this value, however, VTCS also considers where fragmented space occurs. For example, if the first fragmented space is near the end of the MVC, VTCS may process the MVC before an MVC with more total fragmented space.



Hint: You can use Query to display the MVCs eligible for reclamation in your MVC pool. You can also use Query to display information about MVC status and space.

Eligible/Total MVCs Threshold - Starts Automatic Space Reclamation. This policy specifies a percentage value, which is equal to:

$$(Reclaim\ Candidates / Reclaim\ Candidates + Free\ MVCs) * 100$$

Where:

Reclaim Candidates

is the number of Reclaim Candidates determined by the CONFIG RECLAIM THRESHLD parameter. For more information, see “MVC Fragmented Space Threshold- Determines MVC Eligibility for Reclamation” on page 50.

Reclaim Candidates + Free MVCs

equals the number of Reclaim Candidates *plus* the number of free MVCs. A free MVC:

- Has 100% usable space and does not contain any migrated VTVs.
- Is defined as described in “Define Available MVCs with CONFIG” on page 23 and “Define the MVC Pool” on page 24.
- Is writeable.
- Is resident in the ACS.

Use the CONFIG RECLAIM START parameter to set this policy. Valid values for the START parameter are 1 to 98%. The default is 35%.

For each ACS (not globally for all ACSs), VTCS issues message SLS6616I and starts an automatic space reclamation if *both* of the following occurs:

- The actual value of $(Reclaim\ Candidates / Reclaim\ Candidates + Free\ MVCs) * 100$ exceeds the value specified on CONFIG RECLAIM START parameter.
- The number of eligible MVCs exceeds the value specified on the MAXMVC parameter; for more information, see “Maximum MVCs Processed Per Reclaim” on page 52.



Note: The only exception to the above two conditions occurs if an SLS6699 message indicates a critical shortage of free MVCs, in which case automatic reclamation will start anyway.

The following are general guidelines for specifying values for the START parameter:

- A *low* value (for example, 5%), starts automatic space reclamation when there are *few* eligible MVCs compared to free MVCs *unless* you set the MAXMVC value high compared to the number of eligible MVCs.
- A *high* value (for example, 95%), starts automatic space reclamation when there are *many* eligible MVCs compared to free MVCs *unless* you set the MAXMVC value *very* high and your MVC pool is *very* small.



Hint: You can use Query MVCPOOL to display eligible and free MVCs.

Maximum MVCs Processed Per Reclaim. Automatic and demand space reclamation processes one MVC at a time. You can, however, control the maximum number of MVCs that will be processed in a single space reclamation run with the CONFIG RECLAIM MAXMVC parameter. You can use RECLAIM to change the CONFIG setting for the MAXMVC parameter.



For automatic space reclamation to start via the CONFIG RECLAIM START parameter setting, the number of eligible MVCs (determined by the CONFIG RECLAIM THRESHLD parameter) must also exceed the MAXMVC value. For more information, see “Eligible/Total MVCs Threshold - Starts Automatic Space Reclamation” on page 51.

The following are general guidelines for specifying values for the MAXMVC parameter:

- A *low* value reclaims *fewer* MVCs in a single run, but may have *negligible* effect on migrations and recalls and may start automatic space reclamation *more* frequently; for more information, see “Eligible/Total MVCs Threshold - Starts Automatic Space Reclamation” on page 51.
- A *high* value reclaims *more* MVCs in a single run, but may have *considerable* effect on migrations and recalls and may start automatic space reclamation *less* frequently; for more information, see “Eligible/Total MVCs Threshold - Starts Automatic Space Reclamation” on page 51.



Hint: You can use Query to display eligible and free MVCs.

Chapter 2. Preparing for Installation

Before doing the tasks described in Chapter 3 “Installing HSC and VTCS”, complete the preparation tasks described in the following sections:

- “Defining A Security System User ID for HSC, SMC, and VTCS” on page 54
- “Configuring MVS Device Numbers and Esoterics” on page 54
- “Setting the MVS Missing Interrupt Handler (MIH) Value” on page 55
- “Specifying the Region Size” on page 55

Defining A Security System User ID for HSC, SMC, and VTCS

VSM software (HSC, SMC, and VTCS) uses the MVS System Authorization Facility (SAF) to control the usage of MVCs. Before mounting an MVC and before issuing writes to an MVC, VSM issues a SAF query to verify that the HSC user has UPDATE authority for the MVC.

You must define a security system user ID to be used by the SAF queries. All VSM SAF queries are issued on behalf of HSC, so the security system user ID must be associated with the HSC started task. Refer to your security system documentation for details on how to associate a security system user ID with the HSC started task.

You must also set up TAPEVOL profiles to ensure that VSM is authorized to mount MVCs, to ensure that VSM is authorized to update MVCs, and to guard against accidental overwrites of MVCs; for more information, see “Defining MVC Pool Volser Authority” on page 90.



Warning: Depending on the default settings of your security system, VSM may not be able to mount and to write to MVCs until you have defined a security system user ID for HSC and TAPEVOL profiles for the MVCs.

Configuring MVS Device Numbers and Esoterics

The following sections tell how to use the HCD facility to do the following:

- Assign MVS device numbers to VTDs and shared RTDs.
- Associate VTD MVS device numbers and esoterics.

You determined these values in “Determining VSM Configuration Values” on page 9 and recorded them in Table 16. on page 97. See your IBM documentation for more information on the HCD facility.

Assigning MVS Device Numbers to VTDs

Use the HCD facility to assign MVS 3490E device numbers to your VSM system’s VTDs. You determined these device numbers in “VTD Unit Addresses” on page 11. For more information about assigning these device numbers, see *Virtual Storage Manager Planning, Implementation, and Usage Guide*.

Associating VTD MVS Device Numbers and Esoterics

If you use esoteric substitution to allocate VTDs as described in “VSM Esoterics and Esoteric Substitution” on page 12, use the HCD facility to associate each esoteric name with the MVS device numbers for the VTDs that you have chosen for that esoteric.

Assigning MVS Device Numbers to RTDs

Use the HCD facility to assign MVS device numbers to these RTDs.



Hint: You must use the same unit addresses you determined for these transports for UNITATTR statements and LIBGEN updates as described in “RTD Unit Addresses and Identifiers” on page 19.

Setting the MVS Missing Interrupt Handler (MIH) Value

The VTSS's internal error recovery procedures requires the MVS missing-interrupt handler (MIH) value to be 20 minutes. You set this value by modifying the MIH parameter in SYS1.PARMLIB member IECIOSxx.



Adjust applications running on your system that detect missing interrupts and that are independent of the system MIH setting to allow a five-minute MIH value.

Specifying the Region Size

StorageTek recommends that you run HSC/VTCS with a region size of **at least 6 MB**.

Chapter 3. Installing HSC and VTCS

Before you install the software, complete the pre-installation tasks described in the following sections:

- “Reviewing Coexistence Requirements” on page 58
- “Verifying Installation Materials” on page 58

Next, install HSC, HSC maintenance, and VTCS 5.1.0 as described in the following sections:

- “Installing HSC and SMC” on page 60
- “SMC Installation Considerations” on page 60
- “Receiving the VTCS 5.1.0 FMID” on page 61
- “Receiving the VTCS 5.1.0 Service” on page 61
- “Creating the VTCS 5.1.0 LINKLIB Data Sets and Defining Libraries to the HSC Target Zone” on page 62
- “Applying the VTCS 5.1.0 FMID” on page 63
- “Accepting the VTCS 5.1.0 FMID” on page 64
- “Applying the VTCS 5.1.0 Service” on page 65
- “APF Authorizing the VTCS LINKLIB” on page 65
- “Modifying the HSC Startup Procedure to include the VTCS 5.1.0 LINKLIB” on page 65



Caution: In a VSM configuration with multiple hosts that share the same HSC CDS, StorageTek strongly recommends that you install VTCS on all MVS hosts by completing all tasks described in this chapter. Installing VTCS on all MVS hosts ensures that these hosts cannot scratch an MVC.

Reviewing Coexistence Requirements

For more information, see “VTCS System Software Requirements” on page 5.



Hint: If you are upgrading from VTCS 4.0.0 to VTCS 5.1.0, StorageTek recommends that you do the following:

1. Install HSC 5.1.0, which is required for VTCS 5.1.0
2. Install VTCS 5.1.0 on a test LPAR and verify that VTCS 5.1.0 is stable on the test LPAR.
3. Install VTCS 5.1.0 on all hosts and roll into production workloads.
4. Convert the CDS to VSM Extended Format before the next release of VTCS.



Note: The VSM Extended Format CDS **is required for VTCS 5.1.0 or higher with VSM4s that are defined with greater than 64 VTDs and/or greater than 8 RTDs. Also note that** after you convert the CDS to VSM Extended Format, you **cannot** run VTCS 4.0.0 or lower against the converted CDS. For more information, see “Converting the Formatted CDS to VSM Extended Format” on page 71.

Verifying Installation Materials

Before installing VTCS, make sure you have the following installation materials:

- VTCS 5.1.0 product tape
- VTCS 5.1.0 service tape



Note: Contact StorageTek for the latest PTFs to both VTCS 5.1.0 and HSC before you install VTCS.

VTCS Installation Tape Contents

As described in “Verifying Installation Materials” on page 58, VTCS 5.1.0 is delivered on two tapes. The product tape contains the VTCS 5.1.0 FMIDs and the service tape contains PTFs for all VTCS 5.1.0 FMIDs. You **must** install both tapes.

Table 13 lists the files included on the VTCS 5.1.0 product tape.

Table 13. VTCS 5.1.0 Product Tape Contents

File	Data Set Name	Description
1	SMPMCS	SMP/E control statements
2	SWS5100.F1	SWS5100 JCLIN
3	SWS5100.F2	SWS5100 object modules
4	SWS5100.F3	SWS5100 SAMPLIB and MACLIB members (automatically installed in the HSC SAMPLIB and MACLIB)



Note: The VTCS 5.1.0 installation automatically installs the following VTCS members in the HSC SAMPLIB:

SWSJCRDB

Sample CONFIG utility JCL

SWSJMVCR

Sample MVC RPT utility JCL

SWSJVTVR

Sample VTRPT utility JCL

Table 14 lists the files included on the VTCS 5.1.0 service tape.

Table 14. VTCS 5.1.0 Service Tape Contents

File	Data Set Name	Description
1	SYSMODS	SYSMODS

VTCS FMIDs

The VTCS 5.1.0 software is packaged in standard SMP/E format. The VTCS 5.1.0 installation tape includes the following FMID:

SWS5100

The SWS5100 function contains the VTCS load modules for VTCS 5.1.0 running with HSC 5.1.0.



Note: The SWS5100 FMID is a subsidiary HSC 5.1.0 FMID (S0S5100 for HSC 5.1.0) and you must apply the SWS5100 FMID to the same SMP/E zone as HSC.

The VTCS service tape contains SYSMODS (all maintenance since base).

Installing HSC and SMC

Install HSC, SMC and all HSC/SMC maintenance. For more information, see Chapter 5, “Installing the MVS/HSC Functions” of *NCS Installation Guide*.



StorageTek **strongly recommends** upgrading HSC and the LibraryStation and MVS/CSC components (if you intend to connect MVS/CSC clients to VSM) to the current maintenance level before installing VTCS. Also note that StorageTek **requires** you to install SMC, which is required to perform allocation influencing and message interception on MVS.



Note: If you are running at an SMP/E level lower than that supplied with OS/390 Version 2.5, you *must* accept the HSC FMID (S0S5100 for HSC 5.1.0) and all HSC maintenance before installing VTCS 5.1.0.

SMC Installation Considerations

Caution: Ensure that you create and install the SMC JES3 IATUX09 User Exit as described in *NCS Installation Guide*. This exit modification enables deferred mount processing, without which VTV mounts may fail.

Receiving the VTCS 5.1.0 FMID

Use the example JCL in Figure 2 to receive the VTCS 5.1.0 FMID.

```
//S1      EXEC PGM=GIMSMP,
//        PARM='PROCESS=WAIT',
//        DYNAMNBR=120
//SMPCSI  DD DISP=SHR,DSN=hsc.CSI
//SMPPTFIN DD DISP=SHR,DSN=SMPMCS,VOL=SER=WS5100,UNIT=3480,
//        LABEL=(1,SL)
//SMPCTL  DD *
//        SET    BOUNDARY (GLOBAL) .
//        RECEIVE S(SWS5100).
```

Figure 2. JCL Example: Receiving the VTCS 5.1.0 FMID

Receiving the VTCS 5.1.0 Service

Use the example JCL in Figure 3 to receive the VTCS 5.1.0 service.

```
//S1      EXEC PGM=GIMSMP,
//        PARM='PROCESS=WAIT',
//        DYNAMNBR=120
//SMPCSI  DD DISP=SHR,DSN=hsc.CSI
//SMPPTFIN DD DISP=SHR,DSN=SYSMODS,VOL=SER=anyvol,UNIT=3480
//        LABEL=(1,SL),
//        DCB=(LRECL=80,BLKSIZE=7200,RECFM=FB)
//SMPHOLD DD DISP=SHR,DSN=HOLDDATA,UNIT=3480,
//        LABEL=(4,SL),VOL=SER=anyvol,
//        DCB=(LRECL=80,BLKSIZE=7200,RECFM=FB)
//SMPCTL  DD *
//        SET    BOUNDARY (GLOBAL) .
//        RECEIVE SYSMODS HOLDDATA.
```

Figure 3. JCL Example: Receiving the VTCS 5.1.0 service



Note: StorageTek **strongly recommends** that you check the StorageTek Customer Resource Center and download any additional maintenance for VTCS posted since the date of the Service Tape you received. You can execute the RECEIVE step again to receive this additional maintenance.

Creating the VTCS 5.1.0 LINKLIB Data Sets and Defining Libraries to the HSC Target Zone

Use the example JCL in Figure 4 to create the VTCS 5.1.0 LINKLIB (SWSLINK) data sets and to define these data sets and the HSC distribution library (ASLSLINK) to the HSC target zone.



StorageTek has intentionally chosen to use the existing HSC distribution library (ASLSLINK).

```

/*==>>EXPECT RC=4 ON FIRST RUN
//DELETE EXEC PGM=IEFBR14
//SWSLINK1 DD DISP=(MOD,DELETE),
//          SPACE=(TRK,0),
//          DSN=h1q.SWSLINK,
//          UNIT=SYSDA
//ALLOC EXEC PGM=IEFBR14,COND=(0,NE)
//SWSLINK1 DD DCB=(BLKSIZE=23476,RECFM=U),
//          DISP=(NEW,CATLG,DELETE),
//          SPACE=(CYL,(1,1,5)),
//          DSN=h1q.SWSLINK,
//          UNIT=SYSDA
//ALTER1 EXEC PGM=GIMSMP
//SMPCSI DD DISP=SHR,DSN=hsc.CSI
//SMPCNTL DD *
SET BDY(tzone) .
UCLIN .

REP DDDEF(SWSLINK) DATASET(h1q.SWSLINK) SHR .
REP DDDEF(ASLSLINK) DATASET(h1q.ASLSLINK) SHR .

ENDUCL .

```

Figure 4. JCL Example: Creating the SWSLINK data sets and defining libraries to the HSC target zone

Applying the VTCS 5.1.0 FMID

Use the example JCL in Figure 5 to apply the FMID for VTCS 5.1.0.

```
//APPTSOS EXEC PGM=GIMSMP,
//          PARM='PROCESS=WAIT',
//          DYNAMNBR=120
//SMPCSI   DD DISP=SHR,DSN=hsc.CSI
//SMPCNTL  DD *
//          SET    BOUNDARY (tzone).
//          APPLY
//          S(SWS5100)
//          GROUPEXTEND.
```

Figure 5. JCL Example: Applying the VTCS 5.1.0 FMID



Note: Because the VTCS FMID is a subsidiary HSC FMID, SMPE will also apply any VTCS PTFs called out by conditional COREQS in the HSC PTFs already applied to your system. If any of these PTFs have HOLDDATA, you will receive a GIM35965I message for each, and the APPLY will fail. This is not an error condition. If you encounter this condition, please review the individual PTF cover letters, note any additional action(s) to be taken, and repeat the APPLY step with the following parameters:

```
APPLY S(SWS5100) GROUPEXTEND
      BYPASS(HOLDSYSTEM) .
```

Accepting the VTCS 5.1.0 FMID

Use the example JCL in Figure 6 to accept the VTCS 5.1.0 FMID.

```
//APPTSOS EXEC PGM=GIMSMP,
//          PARM='PROCESS=WAIT',
//          DYNAMNBR=120
//SMPCSI   DD DISP=SHR,DSN=hsc.CSI
//SMPCNTL  DD *
//          SET BOUNDARY (dzone).
//          ACCEPT
//          S(SWS5100)
//          GROUPEXTEND.
```

Figure 6. JCL Example: Accepting the VTCS 5.1.0 FMID



Note: If you are running at an SMP/E level lower than that supplied with OS/390 Version 2.5, you *must* accept the HSC FMID and all HSC maintenance before installing VTCS 5.1.0. For more information, see “Installing HSC and SMC” on page 60.

Also note that because the VTCS FMID is a subsidiary HSC FMID, SMPE will also accept any VTCS PTFs called out by conditional COREQS in the HSC PTFs already accepted on your system. If any of these PTFs have HOLDDATA, you will receive a GIM35965I message for each, and the ACCEPT will fail. This is not an error condition. If you encounter this condition, please review the individual PTF cover letters, note any additional action(s) to be taken, and repeat the ACCEPT step with the following parameters:

```
ACCEPT S(SWS5100) GROUPEXTEND
        BYPASS(HOLDSYSTEM) .
```

Applying the VTCS 5.1.0 Service

Use the example JCL in Figure 7 to apply the VTCS 5.1.0 service:

```
//APPTPTF EXEC PGM=GIMSMP,
//          PARM='PROCESS=WAIT',
//          DYNAMNBR=120
//SMPCSI   DD DISP=SHR,DSN=hsc.CSI
//SMPCNTL  DD *
//          SET    BOUNDARY (tzone).
//          APPLY PTFs
//          FORFMID (SWS5100)
//          GROUPEXTEND.
```

Figure 7. JCL Example: Applying the VTCS 5.1.0 service



Note: If any of the PTFs to be applied have HOLDDATA, you will receive a GIM35965I message for each, and the APPLY will fail. This is not an error condition. If you encounter this condition, please review the individual PTF cover letters, note any additional action(s) to be taken, and repeat the APPLY step with the following parameters:

```
APPLY PTFs FORFMID(SWS5100)
GROUPEXTEND
BYPASS(HOLDSYSTEM) .
```

APF Authorizing the VTCS LINKLIB

APF authorize the VTCS LINKLIB (SWSLINK). For more information, see Chapter 3, “Performing Post-Installation Tasks for NCS” of *NCS Installation Guide*.

Modifying the HSC Startup Procedure to include the VTCS 5.1.0 LINKLIB

Use the example JCL in Figure 8 as an example of how to modify the HSC startup procedure to start VTCS. Include the VTCS 5.1.0 LINKLIB (SWSLINK) in the STEPLIB **before** the HSC LINKLIB (SLSLINK).

```
//SLS0      PROC PROG=SLSBINIT
//IEFPROC   EXEC PGM=&PROG,TIME=1440,DPRTY=(7,5),
//          PARM='SSYS(SLS0) E(E086) F(23) M(00)'
//STEPLIB   DD DSN=h1q.SWSLINK,DISP=SHR
//          DD DSN=h1q.SLSLINK,DISP=SHR
```

Figure 8. JCL Example: Modifying the HSC started task to include the SWSLINK library

Chapter 4. Reconfiguring NCS

Before you configure VSM, you must do some or all of the HSC reconfiguration tasks described in the following sections:

- “Updating the HSC LIBGEN” on page 68
- “Verifying the LIBGEN” on page 70
- “Formatting the New CDS” on page 70
- “Converting the Formatted CDS to VSM Extended Format” on page 71
- “Updating the HSC Definition Data Sets” on page 73
- “Creating and Using VSM Management and Storage Classes” on page 74
- “Creating and Using Named MVC Pools” on page 79
- “Updating the HSC PARMLIB Member (SLSSYSxx)” on page 82
- “Connecting MVS/CSC Clients to VSM” on page 84
- “Connecting Non-MVS/CSC Clients to VSM” on page 86
- “Restarting NCS” on page 87



Hint: Except for updating the HSC definition data sets, you can do all these tasks before you install VTCS. Most of the tasks in this Chapter require you to specify VSM system values that you determined on page 9 and recorded in Table 16. on page 97.

Also note that if you are upgrading from a previous release of VTCS, you may not need to do all the tasks in this chapter. For example, if you are not adding RTDs to your configuration, you do not need to update the HSC LIBGEN.

Updating the HSC LIBGEN

If your system's RTDs are new transports, you must update the HSC LIBGEN by adding a SLIDRIVS macro to define the device addresses you determined in "RTD Unit Addresses and Identifiers" on page 19.



To update the HSC LIBGEN to define new transports that are RTDs:

1. **Run the HSC Database Decompile (LIBGEN) Utility to create LIBGEN macro statements from your existing CDS.**

Do *not* edit the original LIBGEN, because if the SET Utility was used to change the library configuration stored in the CDS, the original LIBGEN no longer matches the CDS. For more information about the Database Decompile Utility, see Chapter 5, "Utility Functions" of *HSC System Programmer's Guide for MVS*.

2. **After you run the HSC Database Decompile Utility, add a SLIDRIVS macro to define the RTD device addresses.**

You may also need to update related LIBGEN macros, such as the SLIACS, SLILSM and SLIDLIST macros. For more information about the LIBGEN macros, see Chapter 4, "Creating the Library Configuration File (LIBGEN)" of *HSC Configuration Guide for MVS*.



Note: You can specify that Nearline transports can only be used as RTDs on the SLIACS macro as shown below:

```
SLIACS VSMONLY=YES,ACSDRV=(esoteric0,...,esoteric15),LSM=(....)
```

As shown in this example, in the HSC SLIACS macro:

- The VSMONLY=YES parameter specifies that the RTDs in the ACS are attached only to a VSM system for one or more of the HOSTs connected to this ACS. VSMONLY=NO is the default.
- The ACSDRV parameter specifies the esoteric name of each host that refers to the transports attached to this ACS. A comma is a placeholder for any esoteric name not specified **except** when VSMONLY=YES is specified. In this case, the esoteric "placeholder" is ignored (set to blanks), which then "dedicates" the ACS to VSM use. If VSMONLY=NO, then the esoteric of the first host system is substituted when an esoteric is omitted for a host.

Also note that you can use the HSC SET ACS utility to you specify that Nearline transports can only be used as RTDs as follows:

- The ACSDRV parameter specifies the esoteric name of the host that refers to the transports attached to this ACS. A () specifies that the esoteric for the specified host and ACS is set to blank only when VSMONLY (YES) is specified on the SET ACS utility, which then “dedicates” the ACS to VSM use. If VSMONLY (NO), then the esoteric of the first host system is substituted when an esoteric is omitted for a host.
- The VSMONLY parameter is added. VSMONLY (YES) specifies that the RTDs in the ACS are attached only to a VSM if the ACSDRV parameter specifies () for the esoteric for this host. VSMONLY=NO is the default.

Finally, also note that if you want to have MVS/CSC clients connected to an ACS whose RTDs are dedicated to VSM:

- In the LIBDEV startup parameter, for an ACS that is only attached to VSM, the position ACS esoteric name must be blank.
- Omit the LIBUNIT startup parameter for an ACS that is only attached to VSM.
- Omit the UNITMAP startup parameter for an ACS that is only attached to VSM if *both* of following are true:
 - Clients use the same MVS device numbers defined for this ACS’s RTDs.
 - Your configuration has no cartridge tape UCB defined with an MVS device number that matches a device number defined for this ACS’s RTDs.

If either or both of these statements **is not** true, then you must create both a LIBUNIT and a UNITMAP statement to map the drives to the client addresses.

3. After you update the LIBGEN macros, reassemble and link–edit the LIBGEN file.

For more information, see “LIBGEN Process Verification” in Chapter 4, “Creating the Library Configuration File (LIBGEN)” of *HSC Configuration Guide for MVS*.

Verifying the LIBGEN

After you assemble and link edit the LIBGEN file, run the SLIVERFY program to verify the LIBGEN. For more information, see “Verifying the Library Generation” in Chapter 4, “Creating the Library Configuration File (LIBGEN)” of *HSC Configuration Guide*.



Caution: Before you run SLIVERFY, if your system’s RTDs are new transports that you will share with MVS, you must install them and define their MVS unit addresses via the HCD facility as described in “Assigning MVS Device Numbers to RTDs” on page 54.

Formatting the New CDS

In “HSC CDS DASD Space” on page 38, you determined the size of the new CDS to support your VSM system. After you verify the LIBGEN, you must format a new CDS to this size by using the HSC SLICREAT macro. For more information on the HSC SLICREAT macro, see Chapter 5, “Initializing the Control Data Sets” of *HSC Configuration Guide*.



Note that:

- Before converting the CDS to VSM Extended Format as described in “Converting the Formatted CDS to VSM Extended Format” on page 71, you need to allocate a new data set for the VSM Extended Format CDS.
- If you change the CDS data set name, make sure to update the name in the HSC started task and in any other started tasks or batch jobs (such as Expr and ExLM) that reference this data set.

Converting the Formatted CDS to VSM Extended Format

HSC 5.0.0 and above lets you convert the CDS to VSM Extended Format to support future enhancements to VTCS. **Note that the VSM Extended Format CDS is required for VTCS 5.1.0 or higher with VSM4s that are defined with greater than 64 VTDs and/or greater than 8 RTDs. Also note that** after you convert the CDS to VSM Extended Format, you **cannot** run VTCS 4.0.0 or lower against the converted CDS.

Depending on your site's requirements and schedules, you can upgrade NCS and convert the CDS as follows:

- You can upgrade from either HSC 4.0 or HSC 5.0 to HSC 5.1. Once at HSC 5.1, you can reformat the CDS to VSM Extended Format at a convenient time, and then upgrade to subsequent releases of HSC when desired.
- You can migrate from HSC 4.0 to HSC 5.0 or from HSC 4.0 to HSC 5.1, then convert the CDS.
- You can also upgrade from HSC 4.1 to HSC 5.0 or from HSC 4.1 to HSC 5.1, then convert the CDS.

Figure 9 shows HSC CDS format compatibility across the supported HSC releases.

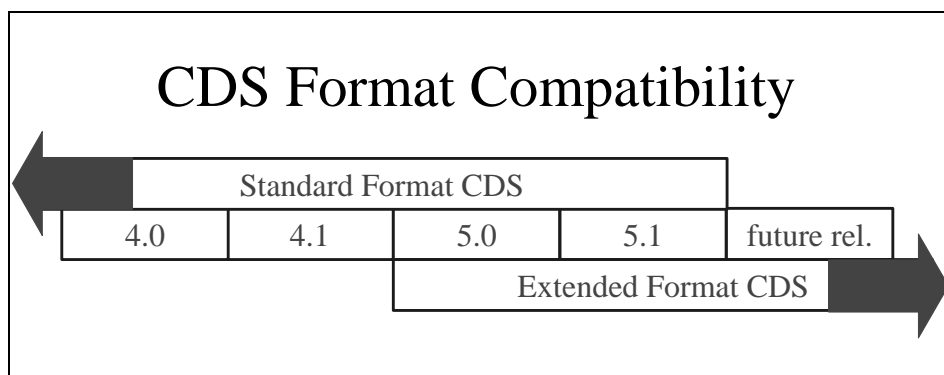


Figure 9. CDS Format Compatibility



Caution: VTCS 4.0.0 and below is **incompatible with** and **will not initialize with** the VSM Extended Format CDS.



To convert the CDS to VSM Extended Format:

1. **Run the HSC SLICREAT macro to format the CDS.**

You can use HSC SLICREAT 4.0.0 and above for this task. For more information, see “Formatting the New CDS” on page 70.



Note:

- This is a newly allocated data set, **not** the 4.0.0 CDS.
- You **cannot use** HSC 5.0.0 CD EXPAND to format the CDS for this procedure.

2. Run the DECOM utility to list your current VSM configuration.

You can use VTCS DECOM 4.0.0 and above for this task.

3. Specify CONFIG VER(NEW) on the DECOM listing and any other desired changes as shown in Figure 10.

Also ensure that you specify the new data set you allocated Step 1 for the VSM Extended Format CDS.

```
//CREATECFG EXEC PGM=SWSADMIN,PARM='MIXED'
//STEPLIB DD DSN=h1q.SLSLINK,DISP=SHR
//SLSCNTL DD DSN=FEDB.VSMLMUL5.DBASEPRM,DISP=SHR
//SLSPRINT DD SYSOUT=*
//SLSIN DD *
CONFIG VER(NEW)
.
.
.
(modified CONFIG statements from DECOM)
.
.
.
```

Figure 10. CONFIG example: VER(NEW) specified to prepare to convert the CDS to VSM Extended Format

4. Run the CONFIG utility, using the modified DECOM listing from Step 3 as input.

You **must use** VTCS CONFIG 5.0.0 and above for this task.

5. Run HSC 5.0 MERGECDs with the ALL parameter to copy and convert the CDS to VSM Extended Format as shown in Figure 11.

```
//MERGECDs EXEC PGM=SLUADMIN,PARM='MIXED'
//SLSFCTL DD DSN=FEDB.FROMVM4.DBASEPRM,DISP=SHR
//SLSFCTL2 DD DSN=FEDB.FROMVM4.DBASESEC,DISP=SHR
//SLSFSTBY DD DSN=FEDB.FROMVM4.DBASESBY,DISP=SHR
//SLSPRINT DD SYSOUT=*
//SLSIN DD *
MERGECDs ALL
/*
```

Figure 11. MERGECDs example: converting the CDS to VSM Extended Format



Note: As shown in Figure 11, in the MERGECDs JCL you specify the **old** CDS. MERGECDs senses an old format CDS and automatically converts it to VSM Extended Format. You **must use** HSC MERGECDs 5.0.0 and above for this task.

Also note that HSC must be at Full Service Level with no tape activity for the MERGECDs VSM Extended Format conversion to succeed. If HSC is only at Base Service Level the VTCS records **are not** copied!

Updating the HSC Definition Data Sets

After you format the new CDS, update the HSC definition data for VSM as described in the following sections.

Creating VTD UNITATTR Statements

For each VTSS in the VSM configuration, create a separate UNITATTR statement that specifies the following:

- On the ADDRESS parameter, the unit address range of the VTDs for that VTSS that you determined in “VTD Unit Addresses” on page 11.
- On the VTSS parameter, the VTSS identifier that you determined in “VTSS Identifiers” on page 10.
- Virtual on the MODEL parameter.

Creating RTD UNITATTR Statements

RTDs, which are Nearline transports, require UNITATTR statements. If your RTDs are new transports, create UNITATTR statements to define the attributes of all the RTD transports to HSC. For more information, see Chapter 3, “Control Statements and Start Procedure” of *HSC System Programmers Guide*.

Creating MVC VOLATTR Statements

MVCs require VOLATTR statements to ensure that VTCS will select the correct RTD device type for each MVC. Create VOLATTR statements for any new volumes you will use as MVCs. For more information, see Chapter 3, “Control Statements and Start Procedure” of *HSC System Programmers Guide*.

Creating an MVC Pool

This procedure tells how to create a pool of volumes that VSM uses as MVCs.



To create an MVC pool:

1. **Edit the data set that will contain your system’s MVCPOOL statements.**
2. **Add MVCPOOL statements for the MVC ranges and save the data set.**
3. **Run the VT MVCDEF command to activate the updated data set.**



Hint: MVCs require additional definitions besides the MVCPOOL statements. For more information, see “MVC Definitions” on page 20.

Creating and Using VSM Management and Storage Classes

VTCS 4.0 and above provides VSM policy management through VSM Management and Storage Classes. See the following sections for more information:

- “Basic Procedure for Creating and Using VSM Management and Storage Classes”
- “Using Storage and Management Classes to Group Multiple Workloads on Shared MVCs”
- “Using Storage and Management Classes to Segregate Individual Workloads on Separate Sets of MVCs”
- “Using Storage and Management Classes to Specify the Source and Target MVC for VTV Consolidation”



Caution: Note the following:

- Use only the minimum Storage Classes required to define the policies you want to implement. Excessive Storage Classes can impact VSM performance due to the MVC mount/dismount overhead incurred. In addition, an MVC can only contain VTVs in a single Storage Class, so excessive Storage Classes can underuse MVC space.
- **If you decide to delete a Management Class definition**, run a VTV Report to make sure that the Management Class is no longer assigned to any VTVs, otherwise unpredictable results will occur!

Basic Procedure for Creating and Using VSM Management and Storage Classes



The following steps show the basic procedure for creating and using VSM Management and Storage Classes.

To create and use VSM Management and Storage Classes:

1. **Determine the definition data set that contains the STORclas and MGMTclas statements.**

As described in “HSC Definition Data Set Names” on page 9, MGMTclas and STORclas statements must reside in the same data set for cross-validation.

2. **Enable the VSM Advanced Management Feature via the HSC FEATURES control statement.**
3. **Define Storage Classes via the STORclas control statement.**
4. **Define Management Classes with the MGMTclas control statement.**

Note that the MGMTclas control statement specifies Storage Classes on the MIGPOL, CONSRC, and CONTGT parameters.

5. **Load the MGMTclas and STORclas control statements with the MGMTDEF command.**
6. **Specify the Management Class name to VTCS on any of the following:**
 - The HSC TAPEREQ statement.
 - The MVS/CSC TAPEREQ statement.
 - SMS routines that you write to the StorageTek DFSMS interface; for more information, see “The StorageTek DFSMS Interface” on page 94.



Note: If you specify a Management Class on a TAPEREQ statement and an SMS routine, the Management Class on the SMS routine takes precedence.

If VTCS receives a request to migrate a VTV that is assigned to an invalid Management Class, VTCS will dynamically create the !ERROR Storage Class and migrate the VTVs defined by the invalid Management Class to the !ERROR Storage Class. MVC reports show when a VTV is migrated to this Storage Class.

Using Storage and Management Classes to Group Multiple Workloads on Shared MVCs

You can use Storage and Management Classes to group multiple workloads on a shared set of MVCs. For example, the `STORCLAS` statements in Figure 12 define Storage Classes `LOC`, `LOCAC`, `REM`, and `REMAC`.

```
STORCLAS NAME(LOC) ACS(00) MEDIA(ECART,ZCART,STK1R)
STORCLAS NAME(LOCAC) ACS(00) MEDIA(ECART,ZCART,STK1R)
STORCLAS NAME(REM) ACS(01) MEDIA(STK1R)
STORCLAS NAME(REMAC) ACS(01) MEDIA(STK1R)
```

Figure 12. Storage Classes for Workload Grouping

Figure 13 defines the following Management Classes:

- Management Classes `PAY` and `ACCOUNT` both specify the `LOCAC` and `REMAC` Storage Classes on the `MIGPOL` parameter. VTCS duplexes VTVs in both Management Classes to identical Storage Classes; the VTVs in `PAY` and `ACCOUNT`, therefore, are duplexed and grouped on the MVCs defined by Storage Classes `LOCAC` and `REMAC`.
- Management Class `PROD` specifies the `LOC` and `REM` Storage Classes on the `MIGPOL` parameter. The VTVs in `PROD`, therefore, are duplexed and grouped on the MVCs defined by Storage Classes `LOC` and `REM`.

```
MGMT NAME(PAY) MIGPOL(LOCAC,REMAC)
MGMT NAME(ACCOUNT) MIGPOL(LOCAC,REMAC)
MGMT NAME(PROD) MIGPOL(LOC,REM)
```

Figure 13. Management Classes for Workload Grouping

Figure 14 consists of `TAPEREQ` statements that do the following:

- Data sets with qualifiers of `PAYROLL.**` are routed to VSM, Management Class `PAY`, and data sets with qualifiers of `ACCOUNTS.**` are routed to VSM, Management Class `ACCOUNT`. As shown in Figure 13, these Management Classes specify identical Storage Classes on the `MIGPOL` parameter, so all data sets with qualifiers of `PAYROLL.**` and `ACCOUNTS.**` are duplexed and grouped on the MVCs defined by Storage Classes `LOCAC` and `REMAC`.
- All other data sets are routed to VSM, Management Class `PROD`; these data sets, therefore, are duplexed and grouped on the MVCs defined by Storage Classes `LOC` and `REM`.

```
TAPEREQ DSN(PAYROLL.**) MEDIA(VIRTUAL) MGMT(PAY)
TAPEREQ DSN(ACCOUNTS.**) MEDIA(VIRTUAL) MGMT(ACCOUNT)
TAPEREQ DSN(**) MEDIA(VIRTUAL) MGMT(PROD)
```

Figure 14. TAPEREQ Statements for Workload Grouping

Once an MVC is used for a Storage Class it remains exclusively assigned to that Storage Class while it contains current VTV copies. This grouping of VTVs on MVCs will be retained even after the MVCs undergo reclamation processing.



Caution: You cannot use the default Storage Class (the name of the last VTSS that wrote to the MVC for reclamation or migration) to group workloads.

Using Storage and Management Classes to Segregate Individual Workloads on Separate Sets of MVCs

You can use Storage and Management Classes to segregate individual workloads on separate sets of MVCs. For example, the `STORCLAS` statements in Figure 15 define Storage Classes `LOC`, `CUSTA`, and `CUSTB`.

```
STORCLAS NAME(LOC) ACS(00) MEDIA(ECART,ZCART,STK1R)
STORCLAS NAME(CUSTA) ACS(00) MEDIA(ECART,ZCART,STK1R)
STORCLAS NAME(CUSTB) ACS(00) MEDIA(ECART,ZCART,STK1R)
```

Figure 15. Storage Classes for Workload Segregation

Figure 16 defines the following Management Classes:

- Management Class `CUSTA` specifies the `CUSTA` Storage Class on the `MIGPOL` parameter. VTCS simplexes VTVs in this Management Classes to only the `CUSTA` Storage Class.
- Similarly, VTCS simplexes VTVs in Management Class `CUSTB` to only the `CUSTB` Storage Class, and VTVs in Management Class `PROD` to only the `LOC` Storage Class.

```
MGMT NAME(CUSTA) MIGPOL(CUSTA)
MGMT NAME(CUSTB) MIGPOL(CUSTB)
MGMT NAME(PROD) MIGPOL(LOC)
```

Figure 16. Management Classes for Workload Segregation

Figure 17 consists of `TAPEREQ` statements that do the following:

- As shown in Figure 17, data sets with qualifiers of `CUSTA.**` are routed to VSM, Management Class `CUSTA`, data sets with qualifiers of `CUSTB.**` are routed to VSM, Management Class `CUSTB`, and all other data sets are routed to VSM, Management Class `PROD`. Thus, workload from the `CUSTA.**` data sets is effectively segregated on a separate set of MVCs from the workload from the `CUSTB.**` data sets.

```
TAPEREQ DSN(CUSTA.***) MEDIA(VIRTUAL) MGMT(CUSTA)
TAPEREQ DSN(CUSTB.***) MEDIA(VIRTUAL) MGMT(CUSTB)
TAPEREQ DSN(***) MEDIA(VIRTUAL) MGMT(PROD)
```

Figure 17. TAPEREQ Statements for Workload Segregation



Caution: You cannot use the default Storage Class (the name of the last VTSS that wrote to the MVC for reclamation or migration) to segregate workloads.

Using Storage and Management Classes to Specify the Source and Target MVC for VTV Consolidation

For duplexed VTVs, you can use Storage and Management Classes to specify the source MVC for VTV consolidation. You can also use Storage Classes to specify the target MVC for consolidation. For example, the `STORCLAS` statements in Figure 18 define Storage Classes `LOCFAST`, `LOC9490`, and `REMOTE`.

```
STORCLAS NAME(LOCFAST) ACS(00) MEDIA(ECART,ZCART,STK1R)
STORCLAS NAME(LOC9490) ACS(00) MEDIA(ECART)
STORCLAS NAME(REMOTE) ACS(01) MEDIA(STK1R)
```

Figure 18. Storage Classes for Source and Target MVC for Consolidation

Figure 19 defines the Management Class `PROD` that specifies:

- The `LOCFAST` and `REMOTE` Storage Classes on the `MIGPOL` parameter. VTCS duplexes VTVs in this Management Class to both Storage Classes.
- The `LOCFAST` Storage Class on the `CONSRC` parameter. For consolidations that specify Management Class `PROD`, VTCS consolidates VTV copies from MVCs in the `LOCFAST` Storage Class.
- The `LOC9490` Storage Class on the `CONTGT` parameter. For consolidations that specify Management Class `PROD`, VTCS consolidates VTVs to MVCs in the `LOC9490` Storage Class.

```
MGMT NAME(PROD) MIG(LOCFAST,REMOTE) CONSRC(LOCFAST) CONTGT (LOC9490)
```

Figure 19. Management Class for Source and Target MVC for Consolidation

Figure 20 shows a `TAPEREQ` statement that specifies Management Class `PROD` for all data sets with a qualifier of `PROD.**` to implement the migration and consolidation policies shown in Figure 19.

```
TAPEREQ DSN(PROD.** ) MEDIA(VIRTUAL) MGMT(PROD)
```

Figure 20. TAPEREQ Statement for Source and Target MVC for Consolidation

Figure 21 shows JCL for the `CONSOLID` utility that specifies Management Class `PROD`. For all data sets with a qualifier of `PROD.**`, VTCS selects VTV copies from MVCs in Storage Class `LOCFAST` for the consolidation and consolidates the selected VTVs to MVCs in Storage Class `LOC9490`.

```
CONS MGMT(PROD)
```

Figure 21. CONSOLID Utility JCL Specifying Source and Target MVC for Consolidation by Management Class

Creating and Using Named MVC Pools



Caution: You can use Named MVC Pools to give an application ownership of the MVCs in the named pool. For example, a service bureau might elect to use Named MVC Pools where its customers have legal requirements to buy and own a group of MVCs. If you do *not* have a specific requirement for Named MVC Pools but you *do* want to group or segregate client data on MVCs, StorageTek strongly advises that you do *not* use Named MVC Pools. Instead, use the methods described in the following sections:

- “Basic Procedure for Creating and Using VSM Management and Storage Classes” on page 75
- “Using Storage and Management Classes to Group Multiple Workloads on Shared MVCs” on page 76
- “Using Storage and Management Classes to Segregate Individual Workloads on Separate Sets of MVCs” on page 77
- “Using Storage and Management Classes to Specify the Source and Target MVC for VTV Consolidation” on page 78

The above sections tell how to use Storage Classes to group or segregate data on MVCs that are selected from the system-wide MVC pool. In this case, you need to manage a only a single MVC pool, which includes setting the policies described in “MVC Policies” on page 49 for the entire pool.

If you create Named MVC Pools, you must explicitly manage each pool, which includes ensuring that each pool has sufficient free MVCs and available MVC space, and potentially includes setting different policies for each pool using the `MVCPool` `MVCFREE`, `MAXMVC`, `THRESH`, and `START` parameters.

If you do elect to use Named MVC Pools, use the procedure on page 80 to implement them.



To create and use Named MVC Pools, do the following:

- 1. Edit the data set that contains your system's MVCPool statements.**
- 2. Modify existing MVCPool statements and/or add additional statements to define the Named MVC Pools and save the data set.**

If you do not specify the MVCPOOL NAME parameter, VTCS does not create a Named MVC Subpool and assigns the specified volumes to the default pool (DEFAULTPOOL). You cannot create Named MVC Pools with the reserved names DEFAULTPOOL and ALL.

You can use the optional MVCFREE, MAXMVC, THRESH, and START parameters to specify values for the Named MVC Pool that override the global values specified on CONFIG.

For example, the following MVCPool statement defines volsers 800000 - 804999 as MVCs in Named Pool CUST1POOL with reclamation parameter values that override the CONFIG global values.

```
MVCP V(800000 - 804999)NAME (CUST1POOL) MAXMVC=20 THRESH=70 START=35
```



Caution: A Named MVC Pool *must* contain media of the type specified in the corresponding STORCLAS MEDIA parameter (see Step 4); otherwise, VTCS will issue a “no MVCs available” message during MVC selection.

- 3. Run the VT MVCDEF command to activate the updated data set.**
- 4. Enable the VSM Advanced Management Feature via the HSC FEATURES control statement.**

The Advanced Management Feature is required for the Storage Classes you define in Step 5.

- 5. Define Storage Classes and associate them with Named MVC Pools.**

For example, the following STORCLAS statement defines STORCL1 and associates this Storage Class with Named MVC Pool CUST1POOL. Requests to use MVCs for storage class STORCL1 will result in MVCs being selected only from the named pool CUST1POOL.

```
STOR NAME(STORCL1) MEDIA(ECART,ZCART,STK1R) MVCPOOL(CUST1POOL)
```

- 6. Create Management Classes that specify the Storage Classes you defined in Step 5 and specify these Management Classes when you route data to the Named MVC Pool.**

For more information, see “Basic Procedure for Creating and Using VSM Management and Storage Classes” on page 75.

7. Specify the Management Class name to VTCS on any of the following:

- The HSC TAPEREQ statement.
- The MVS/CSC TAPEREQ statement.
- SMS routines that you write to the StorageTek DFSMS interface; for more information, see “The StorageTek DFSMS Interface” on page 94.



Note: If you specify a Management Class on a TAPEREQ statement and an SMS routine, the Management Class on the SMS routine takes precedence.

If VTCS receives a request to migrate a VTV that is assigned to an invalid Management Class, VTCS will dynamically create the !ERROR Storage Class and migrate the VTVs defined by the invalid Management Class to the !ERROR Storage Class. MVC reports show when a VTV is migrated to this Storage Class.

Updating the HSC PARMLIB Member (SLSSYSxx)

You can specify the VT MVCDEF command as a statement in the HSC PARMLIB. Figure 22 shows an example of UNITDEF, TAPEREQ, the VT MVCDEF command, and MGMTDEF commands specified as statements in the HSC PARMLIB member.

```
UNITDEF DSN(HSC.UNITATTR)
TREQDEF DSN(HSC.TAPEREQ)
VT MVCDEF DSN(VSM.MVCPool)
MGMTDEF DSN(HSC.PARMS)
COMMP METH LMU
FEAT VSM(ADVGMT)
```

Figure 22. Example: Updating the HSC PARMLIB Member for VSM

In Figure 22:

HSC.UNITATTR

is the data set that contains your system's UNITATTR statements (including UNITATTR statements for VTDs).

HSC.TAPEREQ

is the data set that contains your system's TAPEREQ statements (including TAPEREQ statements for VTVs).

VSM.MVCPool

is the data set that contains your system's MVCPool statements.

HSC.PARMS

is the data set that contains your system's MGMTclas and STORclas statements.

COMMP METH LMU

specifies that LMU is the communications method. StorageTek recommends that you specify either LMU or VTAM, not CDS to allow even sharing of resources in a multi-host environment.

FEAT VSM(ADVGMT)

enables the Advanced Management Feature.

Adding SMF Parameters for VTCS to SYS1.PARMLIB

HSC can produce SMF record subtypes for VTCS events. To produce these record subtypes, you must add two statements to your SMF parameters in SYS1.PARMLIB member SMFPRMxx to specify the following:

- HSC subsystem for which records are produced
- Recording interval in seconds
- SMF record subtypes.



Hint: If you use ExPR for VSM reporting, StorageTek recommends that you specify that your system produces the HSC SMF record subtypes 1 through 8 and 10, 11, 13, 14, 15, 16, 17, 18, 19, 20, 21, 25, 26, 27, and 28 as shown in Figure 23.

Figure 23 shows example statements that produce record subtypes 1 through 8 and 10, 11, 13, 14, 15, 16, 17, 18, 19, 20, 21, 25, 26, 27, 28, and 29 at 1500 second intervals for HSC subsystem SLS0.

```
SUBSYS(SLS0,INTERVAL(001500),TYPE(255))
SUBPARM(SLS0(SUBTYPE,
(1-8, 10, 11, 13, 14, 15, 16, 17, 18, 19, 20, 21, 25, 26, 27, 28, 29)))
```

Figure 23. SYS1.PARMLIB member SMFPRMxx example for VTCS SMF records

Connecting MVS/CSC Clients to VSM

The following procedure tells how to update LibraryStation and MVS/CSC 4.0 and above to connect MVS/CSC clients to VSM.



To connect MVS/CSC clients to VSM:

1. Define a virtual ACS using the LibraryStation VIRTACS statement.

For example, to define virtual ACS 126 and map it to VTSS VTSS02, create the following VIRTACS statement:

```
VIRTACS ID(126) VTSSNAME(VTSS02)
```

This concludes this procedure if *both* of following are true:

- Clients use the same MVS device numbers defined for VSM as described in “VTD Unit Addresses” on page 11.
- Your configuration has no cartridge tape UCB defined with an MVS device number that matches a device number defined for VSM.

Otherwise, continue with Step 2 to map the MVS device numbers defined for VSM to client device numbers.

2. For each VTSS to which MVS/CSC clients connect, define VTD device addresses using the MVS/CSC LIBUNIT statement.



Caution: Each device in the LIBUNIT statement must be represented by a UCB and may not be used to access any device other than the associated VTD. These device addresses do not, however, have to be online.

For more information about the LIBUNIT statement, see Chapter 3, “Defining MVS/CSC Startup Parameters” of *MVS/CSC Configuration Guide*.

For example, for a single VTSS (VSM2 or VSM3), create a LIBUNIT statement such as the following:

```
LIBUNIT (B00,B01,B02,B03,B04,B05,B06,B07, -  

        B08,B09,B0A,B0B,B0C,B0D,B0E,B0F, -  

        .  

        .  

        B38,B39,B3A,B3B,B3C,B3D,B3E,B3F)
```


3. For each VTSS to which MVS/CSC clients connect, map the VTD device addresses from Step 2 on page 84 to the VTD virtual ACS locations using the MVS/CSC UNITMAP statement.

For more information about the UNITMAP statement, see Chapter 3, “Defining MVS/CSC Startup Parameters” of *MVS/CSC Configuration Guide*.

See Table 11. on page 26 for information on VTD locations in virtual ACSs.

For example, for the device addresses you defined in Step 2, create a UNITMAP statement such as the following:

```
UNITMAP (B00,7E:00:1:0,B01,7E:00:1:1, -
        B02,7E:00:1:2,B03,7E:00:1:3, -
        .
        B3E,7E:03:4:2,B3F,7E:03:4:3)
```



Note: The VIRTACS statement specifies virtual ACS IDs in decimal, but the UNITMAP statement specifies these IDs in hexadecimal.

4. If you route data sets to VSM from MVS/CSC using esoteric substitution, specify the desired VSM esoteric on the MVS/CSC LIBDEV startup parameter.

For example, if you have an esoteric that represents all VTDs in your system (**VIRTUAL**), you specify the following on the LIBDEV parameter:

```
LIBDEV(VIRTUAL)
```



Caution: As shown in this example, StorageTek recommends that you **only** specify a virtual esoteric. **Do not** specify multiple real and virtual esoterics, nor do you specify an esoteric that contains virtual and real devices. Otherwise, unpredictable results can occur!

For more information about VSM esoterics, see “VSM Esoterics and Esoteric Substitution” on page 12. For more information about the LIBDEV parameter, see *MVS/CSC Configuration Guide*.

Connecting Non-MVS/CSC Clients to VSM

The following procedure tells how to connect non-MVS/CSC 4.0 and above clients to VSM and define LibraryStation subpools that contain VTVs. Contact StorageTek Software Support for information on the supported clients.



To connect non-MVS/CSC clients to VSM:

1. Define a virtual ACS using the LibraryStation VIRTACS statement.

For example, to define virtual ACSs 126 and 125 and map them to VTSSs VTSS01 and VTSS02, create the following VIRTACS statements:

```
VIRTACS ID(126) VTSSNAME(VTSS01)
VIRTACS ID(125) VTSSNAME(VTSS02)
```

2. Define an HSC subpool that contains VTVs.



Note: If you define this subpool as ANSI label, VTCS will mount scratch VTV as ANSI labeled tapes from this subpool.

For more information, see Chapter 3, “HSC Control Statements and HSC Start Procedure” of *HSC System Programmer's Guide for MVS*.

3. Define a LibraryStation subpool that corresponds to the HSC subpool in Step 2 using the LibraryStation SPNUM statement.

For example, create the following SPNUM statement to:

- Define VTV subpool 7 that corresponds to HSC subpool LSVIRT1
- Pass Management Class MGMTCLS7 to VSM when a VTV is mounted
- Specify VTSSs VTSS01 and VTSS02 are used to satisfy VTV mounts
- Restrict VTV mount requests to the client at IP address 129.80.57.16

```
SPNUM NUM(07) SPNAME(LSVIRT1) VIRT(YES) MGMT(MGMTCLS7)
VTSSL(VTSS01,VTSS02) IPADDR(129.80.57.16)
```

Restarting NCS

To complete the NCS reconfiguration, start one or more of the following on all hosts that are using the new CDS data sets:

- HSC; for more information, see Chapter 7, “Initializing the HSC” of *HSC Configuration Guide*. If you modified the HSC startup procedure as described in “Modifying the HSC Startup Procedure to include the VTCS 5.1.0 LINKLIB” on page 65, HSC initialization automatically starts VTCS.
- LibraryStation; for more information, see Chapter 4, “Starting and Stopping LibraryStation” of *LibraryStation Operator and System Programmer’s Guide*.
- MVS/CSC; for more information, see Chapter 2, “Operating the MVS/CSC” of *MVS/CSC Operator’s Guide*.

Chapter 5. Configuring VSM

After you reconfigure HSC, you must configure VSM, which includes the tasks described in the following sections:

- “Defining VSM Security”
- “Configuring VTCS” on page 91
- “Running the HSC MERGECDS Utility” on page 92
- “Updating HSM” on page 93
- “Updating the Tape Management System” on page 92
- “Routing Data Sets to VSM” on page 93
- “Starting VTCS” on page 96



Hint: Several tasks in this Chapter require you to specify VSM system values that you determined on page 9 and recorded in Table 16. on page 97.

Defining VSM Security

The following sections tell how to define security for VSM:

- “Defining MVC Pool Volser Authority”
- “Defining VTCS Command Authority” on page 91

Defining MVC Pool Volser Authority

When VSM needs to mount an MVC and to write to an MVC, a SAF query is issued to verify that the HSC user (see “Defining A Security System User ID for HSC, SMC, and VTCS” on page 54) has UPDATE authority for the MVC. The SAF query is issued on behalf of HSC and passed to the system security product (such as RACF, CA-ACF2, or CA-Top Secret).

VSM requires UPDATE authority for the volsers in the MVC pool. All other users should have an access of NONE for these volsers. Similarly, VSM should not have UPDATE authority for any volsers that are not in the MVC pool. See the documentation for your security product for procedures to add the appropriate TAPEVOL security for VSM. Table 15 summarizes these definitions.

Table 15. Security Class, Resource Class, and Access Values for MVC Pool Volser Authority

Class	Resource Name	Recommended User Access Levels
TAPEVOL	MVC Pool Volume Serials	UPDATE - allows VSM to write on MVC

Figure 24 shows an example of a RACF profile and permissions commands to give the user ID VSM8HSC update access to MVC volser CVC024.

```

*****
* Define a profile in the TAPEVOL class for MVC CVC024                                *
*****
RDEFINE TAPEVOL CVC024 UACC(NONE)
*****
** Allow user ID VSM8HSC update access to MVC CVC024                                *
*****
PERMIT CVC024 CLASS(TAPEVOL) ACCESS(UPDATE) ID(VSM8HSC)
*****

```

Figure 24. Example RACF MVC volser access file

Caution: Note the following:



- To ensure that MVCs are not accidentally overwritten, for each MVC volser, you must update your TAPEVOL security as described above and your tape management system. For more information, see “Updating the Tape Management System” on page 92.
- You must also run the HSC UNSCratch Utility to unscratch any current scratch cartridges in the MVC range. For more information, see Chapter 5. “Utility Functions” of the *HSC System Programmer’s Guide for MVS*.
- Depending on the default settings of your security system, VSM may not be able to mount and to write to MVCs until you have defined a security system user ID for HSC and TAPEVOL profiles for the MVCs.
- If you add new ranges of MVCs to your VSM system, remember to update the TAPEVOL profiles to include the new ranges.

Defining VTCS Command Authority

If HSC user exit SLSUX15 sets a return code of UX15CHKA, the exit issues a command authorization request to the system security product. Table 25 shows an example of RACF profile and permissions commands to give user SAM15 access to all VTCS commands (those with a VT command prefix). Note that you can only give a user access to *all* VTCS commands; you cannot give access to individual VTCS commands. For more information, see Chapter 9. “User Exits” of the *HSC System Programmer’s Guide for MVS*.

```
*****
* Define a profile in the OPERCMDS class for all VTCS commands *
*****
RDEFINE OPERCMDS subsysname.VT UACC (NONE)
*****
** Allow user SAM15 update access to all VTCS commands *
*****
PERMIT subsysname.VT CLASS(OPERCMDS) ID(SAM15) ACCESS (UPDATE)
*****
```

Figure 25. Example RACF VTCS command authorization file

Configuring VTCS

You run VTCS CONFIG to define the VSM resources and operational parameters to VTCS.



Caution: You **must** run the VTCS CONFIG utility **before** you run the MERGEcds utility to ensure that the VTCS information is added to the CDS.

Running the HSC MERGECDs Utility

After you run the VTCS CONFIG utility, run the HSC MERGEcds Utility to transfer volume information from the old CDS to the new CDS.

Updating the Tape Management System

To update your tape management system (such as CA-1, CA-Dynam/TLMS, and DFSMSrmm), do the following:

- Add volser ranges for VTVs to your tape management system. Ensure that you do *not* assign vault codes to VTVs.
- Access to the MVCs via an RTD bypasses the MVS intercepts put in place by the tape management system so that it does *not* record within its database any access to the MVCs by VSM and does *not* automatically provide protection against inadvertent overwrites of non-expired data on MVCs. Therefore, if you choose to define MVCs to the tape management system, StorageTek **strongly recommends** that you define them as non-scratch, non-expiring volumes.
- The tape management system requires an entry in the MVS Subsystem Name Table; this entry must precede the entry for HSC. For more information, see Chapter 3 of the *HSC Configuration Guide*.



Note: If you are using AutoMedia for MVS, ensure that VTVs are defined as virtual volumes to direct AutoMedia to bypass DSN checking, which allows AutoMedia to recall, mount, and reuse non-resident scratch VTVs.



Caution: Note the following:

- VTCS has an automatic interface to notify RMM when a VTV becomes scratch, but RMM does not notify VTCS when you unscratch a VTV by changing the CV status to non-scratch. If this is done, you must also run the HSC SLUADMIN utility to unscratch the VTV for VTCS. Otherwise, you may encounter a mount failure when VTCS attempts to select the VTV to service a scratch mount request.
- RMM (DFSMS/RMM) has additional integrity checks at mount time to ensure that the correct volume has been mounted. Because VTCS has features and optimizations that sometimes present a new initialized version of a VTV rather than the current copy of a VTV, it is necessary for VTCS to override these RMM integrity checks. VTCS does these overrides via the LISTVOLUME and CHANGEVOLUME API calls to update the RMM database. You must therefore ensure that HSC has been given the appropriate security access to the RMM API. For more information, see your RMM documentation.

Updating HSM

HSM users that have mixed devices that were “logically” defined as the same type of device, such as 3490E, but are “physically” different, such as T9940, virtual (VTD), or 9490 must set the following parameter in HSM:

```
SETSYS RECYCLEINPUTDEALLOCFREQUENCY(MIGRATION(1))
```

By setting this parameter, when HSM is “recycling”, it will deallocate the input drive after it processes each input tape. This is required where the tapes being recycled are “physically” mixed as described above.

If you do not set this parameter, it is possible that you could allocate a 9490 transport for the first tape, then if the second tape was virtual (VTV) or STK2P, the job would fail due to media incompatibility. That is, you could not physically mount the second tape (virtual or STK2P media) on the 9490 drive that had been allocated for the first tape.

Routing Data Sets to VSM

You recorded your VSM candidate data sets in Table 16. on page 97. To route these data sets to VSM, use any of the techniques described in the following sections:

- The StorageTek DFSMS interface; see page 94.
- HSC or MVS/CSC 4.0 TAPEREQ statements; see page 95.
- HSC or MVS/CSC 4.0 User Exits; see page 95 and page 95.



Note: In addition, you can also change your JCL to direct data sets to VSM although StorageTek does not recommend this method.



Caution: StorageTek strongly recommends that you create VTVs as Standard Label (SL) tapes, otherwise unpredictable results can occur. Also note that VSM does not provide readonly protection for VTVs. That is, even if MVS requests a mount READONLY of a VTV, VSM mounts the VTV as READ/WRITE.

The StorageTek DFSMS Interface

You can use the StorageTek DFSMS interface to route data sets to VSM via Unit Name substitution. These interfaces use names that you code in SMS routines to drive SMC allocation processing. For more information about the StorageTek DFSMS interface, see *SMC Configuration and Administration Guide*.

Use this interface for VSM as follows:

1. Define a Storage Group and Storage Class for data sets to be routed to VSM. The Storage Group must be identical in name to an esoteric that represents VTDs. Use an esoteric name and the data set selections that you recorded in Table 16. on page 97.

For more information on defining and using VSM esoterics for the StorageTek DFSMS interface, see “VSM Esoterics and Esoteric Substitution for the StorageTek DFSMS Interface” on page 13.

2. Code Storage Class and Storage Group ACS routines to assign the correct Storage Class and Storage Groups to virtual tape data when &ACSENVIR = "STKTAP1". For more information, see “Defining DFSMS ACS Routines” in Chapter 2 of *HSC System Programmer's Guide for MVS*.
3. If you want to pass one or more Management Classes to VTCS, create a Management Class that is identical in name to a Management Class you defined on a MGMTclas statement. Then add code to the Management Class ACS routine to assign a VTCS Management Class to selected data sets when &ACSENVIR = "STKTAP1".

Note: The Management Class you define on the MGMTclas statement can specify a VSM Storage Class, which is **not** the same as the ACS Storage Class you define in Step 2.

4. Set the SMC ALLOCDEF or ALLOCJOB command SMS option to ON so the StorageTek DFSMS interface drives SMS ACS routines.



Note: If you specify a Management Class on a TAPEREQ statement and an SMS routine, the Management Class on the SMS routine takes precedence.

For more information about the StorageTek DFSMS interface, see *SMC Configuration and Administration Guide*.

HSC or MVS/CSC TAPEREQ Statements

To route data sets to VSM, you can create either of the following:

- An HSC TAPEREQ statement.
- An MVS/CSC TAPEREQ statement.

To route data sets to VSM with TAPEREQ statements, do one of the following:

- Specify `Virtual` on the `MEDIA`, `MODEL`, or `RECTECH` parameter. If you specify `Virtual`, VSM selects an available VTD in your system and routes the job to that VTD.

In a multi-VTSS environment, therefore, specifying `Virtual` does *not* direct the VTD allocation to a specific VTSS, but lets the allocation occur in any VTSS in the configuration.

- Specify an esoteric that represents VTDs on the `ESOTERIC` parameter. You recorded your VSM esoterics in Table 16. on page 97.

For VSM, esoteric definition and substitution is different in JES2 and JES3. For more information on defining and using VSM esoterics for TAPEREQ statements, see “VSM Esoterics and Esoteric Substitution for TAPEREQ Statements and User Exits” on page 14.

- Specify a scratch subpool that contains virtual volumes.



Caution: Multiple TAPEREQ statements that specify the same or overlapping selection criteria (such as jobname, stepname, or data set) can cause undesirable results (such as assignment of `MEDIA Virtual` *and* an esoteric).

HSC User Exits

To route data sets to VSM with HSC User Exits, do one of the following:

- Use return code `UX02VIRT` (32) in register 15 in HSC User Exit `SLSUX02` (JES2) or for `SLSUX04` (JES3) use `UX04VIRT` (24), which you use to control transport allocation for scratch mounts. To satisfy a scratch mount request, return code `UX0xVIRT` causes VSM to select an available VTD in your system and routes the job to a VTD mounted on that VTD.
- Use esoteric substitution in any of the User Exits that support esoteric substitution. For example, to direct scratch allocation requests to a VTD, specify an esoteric that represents VTDs in the `UX02ESO` field of `SLSUX02` or the `UX04ESOT` field of `SLSUX04`.

For VSM, esoteric definition and substitution is different in JES2 and JES3. For more information on defining and using VSM esoterics for HSC User Exits, see “VSM Esoterics and Esoteric Substitution for TAPEREQ Statements and User Exits” on page 14.

For more information about HSC User Exits, see Chapter 9, “User Exits” of *HSC System Programmer’s Guide for MVS*.

MVS/CSC User Exits

To route data sets to VSM with MVS/CSC User Exits, do one of the following:

- MVS/CSC User Exit `SCSUX02` (JES2 and JES3 without TAPE setup environments), which you use to control transport allocation for scratch

mounts, now supports return code UX02VIRT in register 15. SCSUX04 (JES3 with TAPE setup environment) also supports return code UX04VIRT in register 15. To satisfy a scratch mount request, these return codes cause VSM to select an available VTD in your system and route the data set to a VTV mounted on that VTD.

In a multi-VTSS environment, therefore, these return codes do *not* direct the VTD allocation to a specific VTSS, but let the allocation occur in any VTSS in the configuration.

- Use esoteric substitution in any of the User Exits that support esoteric substitution. For example, to direct scratch allocation requests to a VTD, specify an esoteric that represents VTDs in the UX02ESO field of SCSUX02 or the UX04ESOT field of SCSUX04.

For VSM, esoteric definition and substitution is different in JES2 and JES3. For more information on defining and using VSM esoterics for MVS/CSC User Exits, see “VSM Esoterics and Esoteric Substitution for TAPEREQ Statements and User Exits” on page 14.

For more information about MVS/CSC User Exits, see Chapter 8 or Chapter 9 in *MVS/CSC System Programmer's Guide*.

Starting VTCS

Ensure that you modified the HSC startup procedure as described in “Modifying the HSC Startup Procedure to include the VTCS 5.1.0 LINKLIB” on page 65. HSC initialization automatically starts VTCS, and HSC termination automatically terminates VTCS.

Appendix A. VSM Configuration Record

Table 16 lists the installation and configuration values you determined on page 2. It also provides a record of your site's VSM configuration, which can help you and StorageTek service troubleshoot problems with your VSM system.

Table 16. VSM Configuration Record

Configuration Value	Planning Information	Your Site's Selection
MVCPool data set name	"HSC Definition Data Set Names" on page 9	
MGMTclas/STORclas data set name	"HSC Definition Data Set Names" on page 9	
UNITATTR data set name	"HSC Definition Data Set Names" on page 9	
VOLATTR data set name	"HSC Definition Data Set Names" on page 9	
TAPEREQ data set name	"HSC Definition Data Set Names" on page 9	
VTSS identifiers	"VTSS Identifiers" on page 10	
VTD unit addresses	"VTD Unit Addresses" on page 11	
VSM esoteric names	"VSM Esoterics and Esoteric Substitution" on page 12	
VTV volsers (all)	"VTV Definitions" on page 17	
VTV volsers (scratch pool ranges)	"VTV Definitions" on page 17	
RTD unit addresses	"RTD Unit Addresses and Identifiers" on page 19	
MVC volsers - VOLATTR statements and CONFIG	"MVC Definitions" on page 20	
MVC volsers - MVCPool statements	"MVC Definitions" on page 20	
HSC CDS DASD size	"HSC CDS DASD Space" on page 38	
Tape management system DASD size	"Tape Management System DASD Space" on page 38	
VSM candidate data sets		

	VTSS Policies	
Maximum and minimum concurrent automatic migration, immediate migration, and migrate-to-threshold tasks (CONFIG MAXMIG/MINMIG)	“Maximum and Minimum Concurrent Automatic Migration, Immediate Migration, and Migrate-to-Threshold Tasks” on page 39	
AMT settings	“AMT Settings” on page 40	
Deleting scratched VTVs setting	“Deleting Scratched VTVs” on page 42	
	VTV Migration and Consolidation Policies	
Hosts disabled from migration, consolidation, and export by VTV or Management Class (CONFIG NOMIGRAT)	“Hosts Disabled from Migration, Consolidation and Export by VTV or Management Class” on page 43	
VTV residency interval	“VTV Residency Interval before Automatic Migration Candidacy” on page 44	
Migrate immediate setting (MGMTclas IMMEDMig)	“Immediately Migrate VTVs On Dismount” on page 45	
MVC retain interval	“MVC Retain Interval” on page 46	
VTVs per MVC (CONFIG MAXMVC)	“Maximum VTVs per MVC” on page 46	
ACS and media for migration MVCs	“ACS and Media Type of MVCs for Migration and Reclamation.” on page 46	
VTV duplexing	“VTV Duplexing” on page 46	
Migrate duplexed VTVs to separate ACSs (MGMTclas ACSlist)	“Migrate Duplexed VTVs to Separate ACSs” on page 47	
ACS and media for output MVCs for consolidation	“Output MVC ACS and Media for VTV Consolidation” on page 47	
ACS and media for source MVCs for consolidation	“Source MVC ACS and Media for Consolidation of Migrated Duplexed VTVs” on page 47	
	MVC Space Reclamation Policies	
Hosts disabled from reclamation (CONFIG NORECLAM)	“Hosts Disabled for Reclamation” on page 49	

Free MVCs threshold (CONFIG MVCFREE)	“Free MVCs Threshold - Starts Automatic Space Reclamation” on page 50	
MVC fragmented space threshold (CONFIG THRESH1d)	“MVC Fragmented Space Threshold- Determines MVC Eligibility for Reclamation” on page 50	
Eligible/Total MVCs threshold (CONFIG START)	“Eligible/Total MVCs Threshold - Starts Automatic Space Reclamation” on page 51	
MVCs processed per reclaim (CONFIG MAXMVC)	“Maximum MVCs Processed Per Reclaim” on page 52	

Appendix B. VSM Connectivity Requirements

StorageTek **strongly recommends** full VSM connectivity as explained in this appendix. This appendix also provides examples of the types of processing problems encountered with partial connectivity. “Full VSM connectivity” basically constitutes an all-to-all logical path connection of LPARs to VTSSs that allows VTCS to manage all possible combinations of VTV access and transfer, integrity checking, and so forth. Figure 26 shows a Full Connectivity configuration.

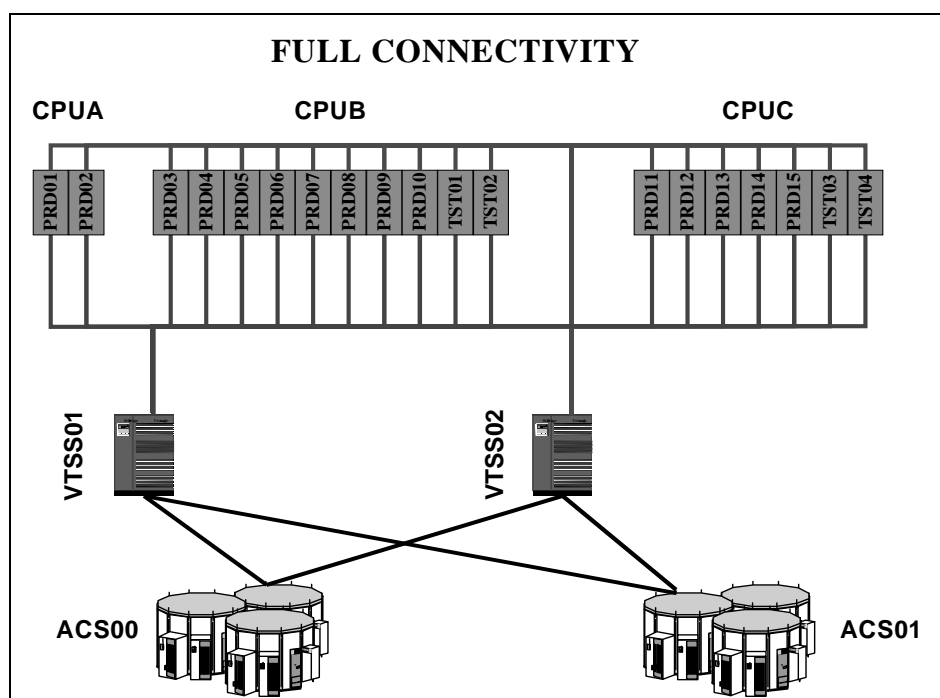


Figure 26. Full VSM Connectivity

Partial Connectivity consists of connections to only those LPARs where throughput or attachment to VTDs are issues, as shown in Figure 27.

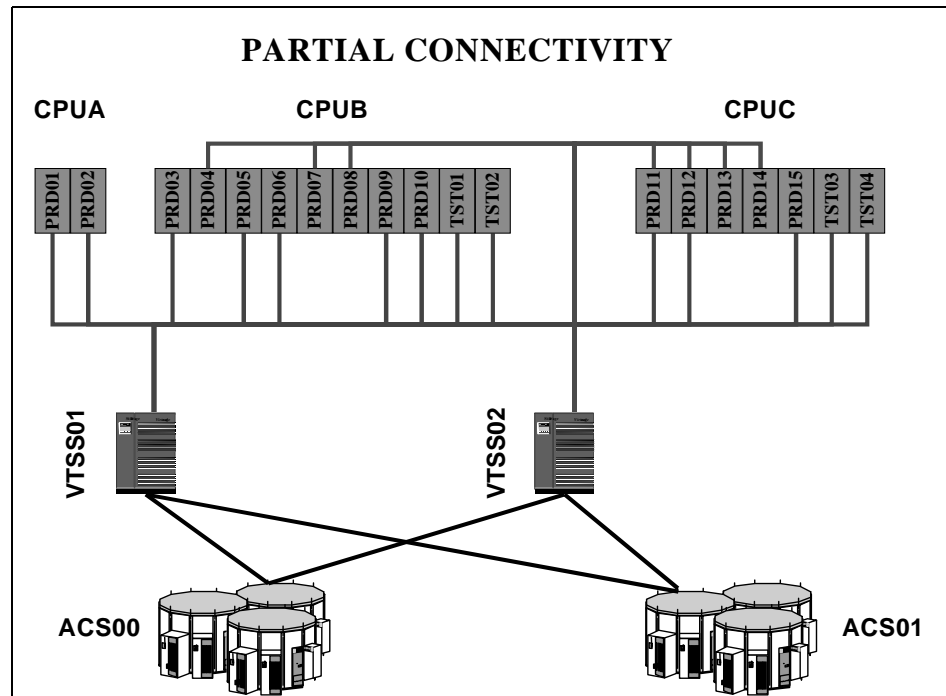


Figure 27. Partial VSM Connectivity

If you have elected to go with Partial Connectivity for throughput, consider the following:

1. If a connection from one LPAR to a VTSS is initially configured, **don't take it away later** because you will generate ECAM activity that cannot complete.
2. Before you initially create a partial configuration, consider the following scenario. For example, in Figure 27:
 - First, PRD04 creates VTV001 on VTSS02.
 - PRD03 (which is **not connected** to VTSS02) then tries to mount VTV001...which will not work because VTCS in PRD03 can't check VTSS02 to ensure that there are no copies of VTV001 resident in that VTSS, which will most likely cause a 522 timeout and job failure.

Therefore, the best way to ensure that you do **not** experience these, and other, problem VSM situations is to configure logical paths for Full Connectivity.

Appendix C. VSM2 and VSM3 Logical Pathing

This appendix consists of the following sections:

- “VSM2 and VSM3 Logical Pathing Overview” on page 104
- “VSM2 and VSM3 Logical Path Planning and Configuration Example” on page 107

VSM2 and VSM3 Logical Pathing Overview

VSM2 and VSM3 logical pathing isn't complex, but you have to understand the following basic concepts:

- You have a **theoretical** maximum of 128 logical paths on the VTSS. However, you must have *some* RTD connections so you cannot allocate *all* 128 logical paths for host-to-VTSS connections. In addition, host-to-VTSS connections are balanced across each port. Therefore, with 16 ports (the most common), each RTD uses a port and a total of 8 logical paths. If you have 8 RTDs, they use 8 ports and 64 logical paths, leaving 8 ports and 64 logical paths for front-end connectivity. 8 ports means a maximum of 8 hosts can connect (if they each use a logical path on every available port), using multiple logical paths if required up to the total of 64. If a ninth host tries to connect...you'll get an error showing no logical paths available.
- Each VTSS has either 4 or 8 ICE cards. Each ICE card has two ports, which are used **either** for host-to-VTSS ESCON channel connections or for VTSS-to-RTD Nearlink connections.
- A logical **host** path is the communication path between a host and **all 64** VTDs within the VTSS. These logical host path(s) also provide host-to-VTSS communications.
- Each VTSS must be connected to a **minimum** of two RTDs in **each** ACS to which the VTSS is connected. VTCS cannot check device type, but StorageTek **strongly recommends** at least two RTDS of each device type in each ACS to which the VTSS is attached....otherwise, you can seriously compromise error recovery and also impact the efficiency of space reclamation. In an 8 ICE card configuration with a single media type, this leaves 14 available ports for host-to-VTSS ESCON channel connections, which equals a **maximum** of 112 logical paths.
- Logical paths to the host(s) are **evenly distributed** across the ICE Cards (and ports) used for host-to VTSS connections, so that:
 - **In an 8-ICE card VTSS**, each ICE card port used for a **physical** host-to-VTSS ESCON channel connection provides **8 logical** paths to the host.
 - **In a 4-ICE card VTSS**, each ICE card port used for a **physical** host-to-VTSS ESCON channel connection provides **16 logical** paths to the host.
- The **actual** host-to-VTSS pathing configuration, therefore, depends on:
 - The number of VTSS ICE Cards installed.
 - The number of RTDs attached to the VTSS.
 - Performance considerations. You should consider assigning more logical paths to hosts requiring performance and redundancy.

In the following sections, let's look at some typical examples of ICE card port configurations and the host logical paths available with each:

- “Host Paths for VTSS with 8 ICE Cards, 4 RTD Nearlink Connections”
- “Host Paths for VTSS with 8 ICE Cards, 8 RTD Nearlink Connections” on page 106
- “Host Paths for VTSS with 4 ICE Cards, 4 RTD Nearlink Connections” on page 106

Host Paths for VTSS with 8 ICE Cards, 4 RTD Nearlink Connections

As Table 17 shows, a VTSS with 8 ICE cards and 4 RTD Nearlink connections supports a maximum of 96 logical host paths.

Table 17. Host Paths for VTSS with 8 ICE Cards, 4 RTD Nearlink Connections

ICE Card Number	ICE00	ICE01	ICE02	ICE03	ICE13	ICE12	ICE11	ICE10	Total Host Paths
1st Port Connection	RTD	RTD	8 host paths	8 host paths	RTD	RTD	8 host paths	8 host paths	32
2nd Port Connection	8 host paths	8 host paths	8 host paths	8 host paths	8 host paths	8 host paths	8 host paths	8 host paths	64
Total Host Paths (Both Cards)									96

Host Paths for VTSS with 8 ICE Cards, 8 RTD Nearlink Connections

As Table 18 shows, a VTSS with 8 ICE cards and 8 RTD Nearlink connections supports a maximum of 64 logical host paths.

Table 18. Host Paths for VTSS with 8 ICE Cards, 8 RTD Nearlink Connections

ICE Card Number	ICE00	ICE01	ICE02	ICE03	ICE13	ICE12	ICE11	ICE10	Total Host Paths
1st Port Connection	RTD	8 host paths	RTD	8 host paths	RTD	8 host paths	RTD	8 host paths	32
2nd Port Connection	8 host paths	RTD	8 host paths	RTD	8 host paths	RTD	8 host paths	RTD	32
Total Host Paths (Both Cards)									64

Host Paths for VTSS with 4 ICE Cards, 4 RTD Nearlink Connections

As Table 19 shows, a VTSS with 4 ICE cards and 4 RTD Nearlink connections supports a maximum of 64 logical host paths.

Table 19. Host Paths for VTSS with 4 ICE Cards, 4 RTD Nearlink Connections

ICE Card Number	ICE00	ICE02	ICE12	ICE10	Total Host Paths
1st Port Connection	RTD	16 host paths	16 host paths	RTD	32
2nd Port Connection	16 host paths	RTD	RTD	16 host paths	32
Total Host Paths (Both Cards)					64

VSM2 and VSM3 Logical Path Planning and Configuration Example

In this example, we connect 3 CPUs, each with multiple LPARS, to two VTSSs. This example builds on the concepts described in “VSM2 and VSM3 Logical Pathing Overview” on page 104.

The planning and configuration of this example consists of four steps:

- “Step 1: Determine Logical Pathing Requirements” on page 108. For each LPAR, we analyze the number of logical paths for each of the following:
 - **Throughput.** In Table 20 on page 110 and Table 21 on page 111, the bandwidth requirements for each LPAR correspond to the number of logical paths we allocate for throughput.
 - **Redundancy.** We allocate redundant logical paths to LPARs that rank high on the “require continuous operations” scale.
 - **Connectivity.** A logical path for connectivity is basically a “yes” answer to the question “Does this LPAR need a connection to a VTSS?” For more information, see Appendix B “VSM Connectivity Requirements”.



Hint: Note the following about Table 20 on page 110 and Table 21 on page 111. Basically, if we allocate any logical paths in the **Throughput** column, we do **not** have to explicitly allocate a logical path for connectivity in the **Add 1 for Connectivity?** column. If we do **not** allocate any logical paths in the **Throughput** column, we **must** explicitly allocate one logical path for connectivity in the **Add 1 for Connectivity?** column.

- “Step 2: Determine Channel Requirements and Allocate Channels” on page 112. The logical path requirements we sized in Step 1 are the input to determining the channel requirements and allocation.
- “Step 3: Allocate Logical Paths” on page 113. Here we overlay the logical paths requirements from Step 1 on top of the channel allocations in Step 2.
- “Step 4: Code The IOCP” on page 122. This is the easy part, thanks to the careful planning we did in Steps 1 through 3. We simply code the IOCP to match the final results, shown in Table 27 on page 116 and Table 32 on page 120.

Step 1: Determine Logical Pathing Requirements

“Logical Pathing Requirements for VTSS01” on page 110 and “Logical Pathing Requirements for VTSS02” on page 111 show examples of allocating logical paths for both VTSSs to satisfy the throughput, connectivity, and redundancy requirements for each LPAR. Now...we said earlier that Logical Pathing really isn’t complex, but it requires the following information to decipher Table 20 on page 110 and Table 21 on page 111. For each **HOST/LPAR** for a specific **CPU**, we have the following information:

- **Bandwidth (MB/Sec)** is the estimated maximum bandwidth required for this **HOST/LPAR**.
- **Number of Front End Paths Required** is derived from its four subcolumns:

- **Throughput** is the number of paths required to provide the estimated maximum bandwidth (**Bandwidth (MB/Sec)**). **Note that** for these examples, we’re assuming a 9 MB/sec bandwidth per path and we’re rounding up.

For example, in Table 20 on page 110, the HOST/LPAR PRD10 on CPUB has a 15 MB/sec throughput requirement, so we round up to 2 paths times 9MB/sec for a total of 18 MB/sec.

- **Redundancy** is the **additional** number of paths required for redundancy. This is, again, an estimate based on the criticality of the application(s) on any particular HOST/LPAR.

Following along with our example, in Table 20 on page 110, the HOST/LPAR PRD10 on CPUB has a mission-critical payroll application, so we give it an one additional path for redundancy.

- **Add 1 for Connectivity?** As we said back in Appendix B “VSM Connectivity Requirements”, configuring for full VSM connectivity is the best way to avoid the kinds of problems that occur with partial connectivity. If you did not allocate any paths for throughput or redundancy, **make sure** that you allocate sufficient paths for connectivity. If, on the other hand, you allocate any paths for throughput/redundancy that **also** ensure full connectivity, you do not need an explicit allocation for redundancy.....so this column is either a 0 or 1.

In Table 20 on page 110, the HOST/LPAR PRD10 on CPUB has plenty of paths allocated for throughput/redundancy, so we don’t need to allocate one for connectivity. PRD04 on CPUA, on the other hand, didn’t get any for throughput or redundancy, so we allocate one for connectivity.

- **Total** is simply the total of the previous three columns. Notice that in the **Total** column you'll see some numbers in **bold**. That means within a CPU, the HOST/LPAR with the highest total paths wins, so that's what we allocate for that CPU. For example, In Table 20 on page 110, for CPUA, HOST/LPAR PRD01 needs a total of 3 paths, and PRD02 needs a total of 4 paths, so we allocate 4 paths for CPUA...maybe.

We say "maybe" because this scheme does not take into account concurrent requirements across the HOST/LPARs within each CPU, so you may want to allow more paths for these requirements. Our calculations do, however, give an indication of the ratio of ports that might be allotted to each CPU. For example, in CPUB, PRD04, PRD07 and PRD08 might each be passing data concurrently, so you might want to allocate something like an additional 5 paths for this activity. As you'll see in "Step 3: Allocate Logical Paths" on page 113, our calculations are rigorous but do not use all available logical paths.

These **bold** numbers in the **Total** column are key to "Step 2: Determine Channel Requirements and Allocate Channels" on page 112, because we plug these numbers into Table 22 on page 112.

Logical Pathing
Requirements for
VTSS01

Table 20 shows logical pathing requirements for VTSS01.

Table 20. Logical Pathing Requirements for VTSS01

CPU	HOST/ LPAR	Bandwidth (MB/Sec)	Number of Front End Paths Required			
			Throughput	Redundancy	Add 1 for Connectivity?	Total
CPUA	PRD01	12	2	1	0	3
	PRD02	34	4	0	0	4
CPUB	PRD03	8	1	1	0	2
	PRD04	0	0	0	1	1
	PRD05	15	2	0	0	2
	PRD06	12	2	0	0	2
	PRD07	0	0	0	1	1
	PRD08	0	0	0	1	1
	PRD09	12	2	0	0	2
	PRD10	15	2	1	0	3
	TST01	5	1	1	0	2
	TST02	5	1	1	0	2
CPUC	PRD11	15	2	0	0	2
	PRD12	12	2	0	0	2
	PRD13	0	0	0	1	1
	PRD14	0	0	0	1	1
	PRD15	12	2	0	0	2
	TST03	5	1	0	0	1
	TST04	5	1	0	0	1

Logical Pathing
Requirements for
VTSS02

Table 21 shows logical pathing requirements for VTSS02.

Table 21. Logical Pathing Requirements for VTSS02

CPU	HOST/ LPAR	Bandwidth (MB/Sec)	Number of Front End Paths Required			
			Throughput	Redundancy	Add 1 for Connectivity?	Total
CPUA	PRD01	0	0	0	1	1
	PRD02	0	0	0	1	1
CPUB	PRD03	0	0	0	1	1
	PRD04	12	2	1	0	3
	PRD05	0	0	0	1	1
	PRD06	0	0	0	1	1
	PRD07	12	2	1	0	3
	PRD08	15	2	1	0	3
	PRD09	0	0	0	1	1
	PRD10	0	0	0	1	1
	TST01	0	0	0	1	1
	TST02	0	0	0	1	1
CPUC	PRD11	12	2	1	0	3
	PRD12	12	2	1	0	3
	PRD13	18	2	1	0	3
	PRD14	18	2	1	0	3
	PRD15	6	1	1	0	2
	TST03	0	0	0	1	1
	TST04	0	0	0	1	1

Step 2: Determine Channel Requirements and Allocate Channels

Using the logical path requirements determined in “Step 1: Determine Logical Pathing Requirements” on page 108, we next determine the channel requirements for each CPU in Table 22. Table 23 then summarizes the actual channel allocations (CHPIDs and connection to VTSS01 or VTSS02).

Table 22. Channel Requirements for Each CPU

CPU	Front-End Channels Required		
	VTSS01	VTSS02	Total
RTDs	6	6	12
CPUA	4	1	5
CPUB	3	3	6
CPUC	2	3	5
Total	15	13	
Note: The total number of front-end channels can be a maximum of 16 per VTSS. If it exceeds 16, go back to Table 21. on page 111 and Table 20. on page 110 and rework to reduce the total logical paths to 16 or less			

Table 23. Channel Allocation for Each CPU

CPU	CHPIDs	VTSS	
		VTSS01	VTSS02
CPUA	06	*	
	2E		*
	4F	*	
	D4	*	
	E3	*	
CPUB	2F	*	
	32	*	
	85	*	
	8C		*
	A4		*
CPUC	C4		*
	0F	*	
	30		*
	58	*	
	88		*
	8D		*

Step 3: Allocate Logical Paths

In this section, the objective is to allocate logical paths to satisfy the throughput/connectivity/redundancy considerations we determined **within the restriction** that we have a maximum of eight paths per port on an 8 ICE card VTSS. We use this allocation to build the Access List and place additional connections in the Candidate List, which we describe in “Step 4: Code The IOCP” on page 122. Note that these are **minimum configurations** for the requirements...in fact, there are logical paths left over that can be allocated if needed.

We'll start with VTSS01, and we'll do this in the following stages:

- “VTSS01 Logical Paths for Throughput”
- “VTSS01 Logical Paths for Throughput and Connectivity” on page 114
- “VTSS01 Logical Paths for Throughput, Connectivity, and Redundancy” on page 115
- “VTSS01 Logical Paths for Throughput, Connectivity, Redundancy, and RTD Connections” on page 116
- “VTSS01 Unallocated Logical Paths” on page 117

VTSS01 Logical Paths for Throughput

Table 24 shows **only** the VTSS01 logical paths required for throughput.

Table 24. VTSS01 Host Logical Paths for Throughput

VTSS Port	CPU	CHPID	Host Logical Paths Allocated							
001	CPUA	06	PRD01	PRD02						
011	CPUC	0F	PRD11	PRD12	PRD15	TST03				
021	CPUB	2F	PRD03	PRD06	PRD09	TST01				
030	CPUA	4F	PRD01	PRD02						
031	CPUB	32	PRD05	PRD06	PRD10	TST02				
101	CPUA	D4	PRD02							
111	CPUC	58	PRD11	PRD12	PRD15	TST04				
130	CPUA	E3	PRD02							
131	CPUB	85	PRD05	PRD09	PRD10					

VTSS01 Logical Paths for Throughput and Connectivity

Table 25 shows the VTSS01 logical paths required for throughput and connectivity (shaded in the table).

Table 25. VTSS01 Host Logical Paths for Throughput and Connectivity

VTSS Port	CPU	CHPID	Host Logical Paths Allocated							
			PRD01	PRD02						
001	CPUA	06	PRD01	PRD02						
011	CPUC	0F	PRD11	PRD12	PRD15	TST03	PRD13			
021	CPUB	2F	PRD03	PRD06	PRD09	TST01	PRD07			
030	CPUA	4F	PRD01	PRD02						
031	CPUB	32	PRD05	PRD06	PRD10	TST02	PRD08			
101	CPUA	D4	PRD02							
111	CPUC	58	PRD11	PRD12	PRD15	TST04	PRD14			
130	CPUA	E3	PRD02							
131	CPUB	85	PRD05	PRD09	PRD10	PRD04				

VTSS01 Logical
Paths for Throughput,
Connectivity, and
Redundancy

Table 25 shows the VTSS01 logical paths required for throughput, connectivity, and redundancy (shaded in the table).

Table 26. VTSS01 Host Logical Paths for Throughput, Connectivity, and Redundancy

VTSS Port	CPU	CHPID	Host Logical Paths Allocated							
001	CPUA	06	PRD01	PRD02						
011	CPUC	0F	PRD11	PRD12	PRD15	TST03	PRD13			
021	CPUB	2F	PRD03	PRD06	PRD09	TST01	PRD07	TST02		
030	CPUA	4F	PRD01	PRD02						
031	CPUB	32	PRD05	PRD06	PRD10	TST02	PRD08	TST01		
101	CPUA	D4	PRD02							
111	CPUC	58	PRD11	PRD12	PRD15	TST04	PRD14			
130	CPUA	E3	PRD02	PRD01						
131	CPUB	85	PRD05	PRD09	PRD10	PRD04	PRD03			

**VTSS01 Logical
Paths for Throughput,
Connectivity,
Redundancy, and
RTD Connections**

Table 27 shows the VTSS01 logical paths required for throughput, connectivity, redundancy, and RTD connections (shaded in the table).

Table 27. VTSS01 Host Logical Paths for Throughput, Connectivity, and Redundancy, and RTD Connections

VTSS Port	CPU	CHPID	Host Logical Paths Allocated							
000	RTD									
001	CPUA	06	PRD01	PRD02						
010	RTD									
011	CPUC	0F	PRD11	PRD12	PRD15	TST03	PRD13			
020	RTD									
021	CPUB	2F	PRD03	PRD06	PRD09	TST01	PRD07	TST02		
030	CPUA	4F	PRD01	PRD02						
031	CPUB	32	PRD05	PRD06	PRD10	TST02	PRD08	TST01		
100	RTD									
101	CPUA	D4	PRD02	PRD01						
110	RTD									
111	CPUC	58	PRD11	PRD12	PRD15	TST04	PRD14			
120	RTD									
121	SPARE									
130	CPUA	E3	PRD02							
131	CPUB	85	PRD05	PRD09	PRD10	PRD04	PRD03			

VTSS01 Unallocated Logical Paths Table 28 shows the VTSS01 unallocated logical paths (shaded in the table).

Table 28. VTSS01 Unallocated Host Logical Paths

VTSS Port	CPU	CHPID	Host Logical Paths Allocated							
000	RTD									
001	CPUA	06	PRD01	PRD02						
010	RTD									
011	CPUC	0F	PRD11	PRD12	PRD15	TST03	PRD13			
020	RTD									
021	CPUB	2F	PRD03	PRD06	PRD09	TST01	PRD07	TST02		
030	CPUA	4F	PRD01	PRD02						
031	CPUB	32	PRD05	PRD06	PRD10	TST02	PRD08	TST01		
100	RTD									
101	CPUA	D4	PRD02	PRD01						
110	RTD									
111	CPUC	58	PRD11	PRD12	PRD15	TST04	PRD14			
120	RTD									
121	SPARE									
130	CPUA	E3	PRD02							
131	CPUB	85	PRD05	PRD09	PRD10	PRD04	PRD03			

Next, we'll allocate logical paths for VTSS02 as follows:

- “VTSS02 Logical Paths for Throughput”
- “VTSS02 Logical Paths for Throughput and Connectivity”
- “VTSS02 Logical Paths for Throughput, Connectivity, and Redundancy” on page 119
- “VTSS02 Logical Paths for Throughput, Connectivity, Redundancy, and RTD Connections” on page 120
- “VTSS02 Unallocated Logical Paths” on page 121

VTSS02 Logical Paths for Throughput

Table 29 shows **only** the VTSS02 logical paths required for throughput.

Table 29. VTSS02 Host Logical Paths for Throughput

VTSS Port	CPU	CHPID	Host Logical Paths Allocated							
011	CPUB	8C	PRD04	PRD07						
021	CPUC	30	PRD11	PRD12	PRD14					
030	CPUB	A4	PRD04	PRD08						
101	CPUC	88	PRD11	PRD13	PRD14					
111	CPUC	8D	PRD12	PRD13	PRD15					
130	CPUB	C4	PRD07	PRD08						

VTSS02 Logical Paths for Throughput and Connectivity

Table 30 shows the VTSS02 logical paths required for throughput and connectivity (shaded in the table).

Table 30. VTSS02 Host Logical Paths for Throughput and Connectivity

VTSS Port	CPU	CHPID	Host Logical Paths Allocated							
001	CPUA	2E	PRD01	PRD02						
011	CPUB	8C	PRD04	PRD07	PRD03	PRD09	TST02			
021	CPUC	30	PRD11	PRD12	PRD14	TST03				
030	CPUB	A4	PRD04	PRD08	PRD05	PRD10				
101	CPUC	88	PRD11	PRD13	PRD14	TST04				
111	CPUC	8D	PRD12	PRD13	PRD15					
130	CPUB	C4	PRD07	PRD08	PRD06	TST01				

VTSS02 Logical
Paths for Throughput,
Connectivity, and
Redundancy

Table 31 shows the VTSS02 logical paths required for throughput, connectivity, and redundancy (shaded in the table).

Table 31. VTSS02 Host Logical Paths for Throughput, Connectivity, and Redundancy

VTSS Port	CPU	CHPID	Host Logical Paths Allocated							
001	CPUA	2E	PRD01	PRD02						
011	CPUB	8C	PRD04	PRD07	PRD03	PRD09	TST02	PRD08		
021	CPUC	30	PRD11	PRD12	PRD14	TST03	PRD13	PRD15		
030	CPUB	A4	PRD04	PRD08	PRD05	PRD10	PRD07			
101	CPUC	88	PRD11	PRD13	PRD14	TST04	PRD12			
111	CPUC	8D	PRD12	PRD13	PRD15	PRD11	PRD14			
130	CPUB	C4	PRD07	PRD08	PRD06	TST01	PRD04			

**VTSS02 Logical
Paths for Throughput,
Connectivity,
Redundancy, and
RTD Connections**

Table 32 shows the VTSS02 logical paths required for throughput, connectivity, redundancy, and RTD connections (shaded in the table).

Table 32. VTSS02 Host Logical Paths for Throughput, Connectivity, and Redundancy, and RTD Connections

VTSS Port	CPU	CHPID	Host Logical Paths Allocated							
000	RTD									
001	CPUA	2E	PRD01	PRD02						
010	RTD									
011	CPUB	8C	PRD04	PRD07	PRD03	PRD09	TST02	PRD08		
020	RTD									
021	CPUC	30	PRD11	PRD12	PRD14	TST03	PRD13	PRD15		
030	CPUB	A4	PRD04	PRD08	PRD05	PRD10	PRD07			
100	RTD									
101	CPUC	88	PRD11	PRD13	PRD14	TST04	PRD12			
110	RTD									
111	CPUC	8D	PRD12	PRD13	PRD15	PRD11	PRD14			
120	RTD									
130	CPUB	C4	PRD07	PRD08	PRD06	TST01	PRD04			

VTSS02 Unallocated Logical Paths Table 33 shows the VTSS02 unallocated logical paths (shaded in the table).

Table 33. VTSS02 Unallocated Host Logical Paths

VTSS Port	CPU	CHPID	Host Logical Paths Allocated							
000	RTD									
001	CPUA	2E	PRD01	PRD02						
010	RTD									
011	CPUB	8C	PRD04	PRD07	PRD03	PRD09	TST02	PRD08		
020	RTD									
021	CPUC	30	PRD11	PRD12	PRD14	TST03	PRD13	PRD15		
030	CPUB	A4	PRD04	PRD08	PRD05	PRD10	PRD07			
031	SPARE									
100	RTD									
101	CPUC	88	PRD11	PRD13	PRD14	TST04	PRD12			
110	RTD									
111	CPUC	8D	PRD12	PRD13	PRD15	PRD11	PRD14			
120	RTD									
121	SPARE									
130	CPUB	C4	PRD07	PRD08	PRD06	TST01	PRD04			
131	SPARE									

Step 4: Code The IOCP



Caution: Ensure that you complete Steps 1 through 3 **before** you code the IOCP, otherwise you may incur unpredictable and undesirable results!

In the IOCP example for VTSS01 in Figure 30 on page 124, we use the input from Table 23 on page 112 to define the following in the PARTITION statement:

- First, the **Access List**, which consists the LPARs that need access to a VTSS.



Note: If more than 8 LPARs are coded in the Access List, the first 8 LPARs that are started will obtain one logical path through this CHPID. The 9th LPAR that is started will get “logical path not available.”

- Next, the **Candidate List**, which consists of any LPARS that might need access to a VTSS some time in the future.

A visual representation of the above text is what’s best at this point, so please see the following graphic that shows the position of the Access List (Figure 28 on page 123) and Candidate List (Figure 29 on page 123) in the PARTITION statement.



Note: Defining Access and Candidate List **is different** between HCD and IOCP. IBM’s *MVS/ESA HCD and Dynamic I/O Reconfiguration Primer* (SG24-4037-01) says, “The HCD candidate list does not contain any partition that is already in the access list. It is viewed as an additional list of partitions that might get access to the channel path at a later time. Thus, a partition defined in the access list of a CHPID does not appear on the Define Candidate List panel. In IOCP, the candidate list includes the access list.”

Completing the assignment of logical paths via the IOCP ensures that **only** those paths that need to be online are brought online and, in addition, all paths that need to be online (for VTCS connectivity purposes) are also online. This completes this example...and we hope it was worth the effort.

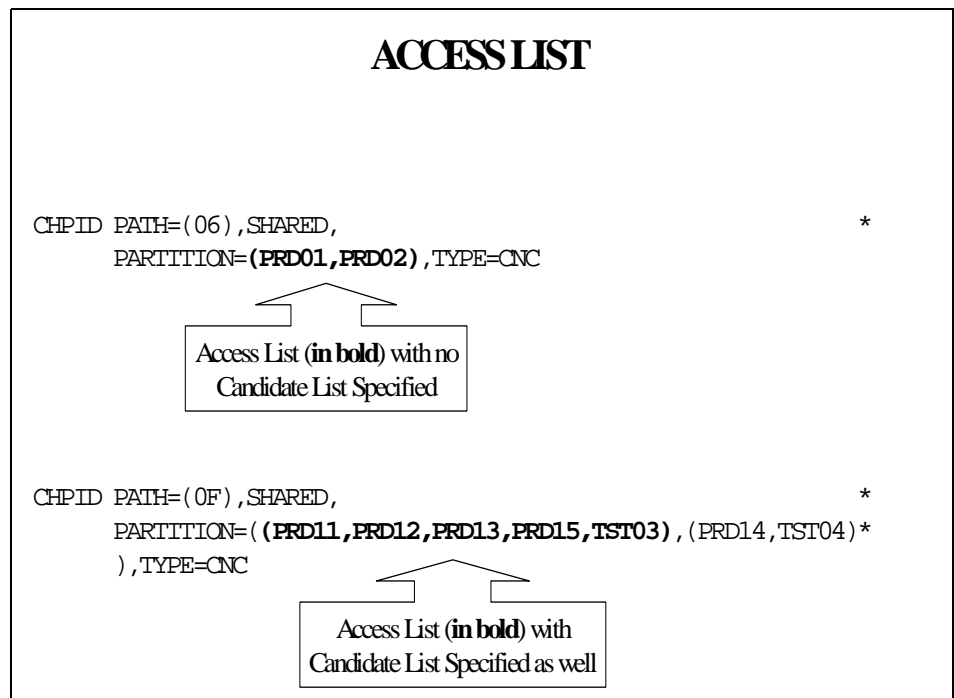


Figure 28. PARTITION Parameter on the CHPID Statement-Access List

Figure 29. PARTITION Parameter on the CHPID Statement-Candidate List

CHPID PATH=(06), SHARED,	*
PARTITION=(PRD01, PRD02), TYPE=CNC	
CHPID PATH=(0F), SHARED,	*
PARTITION=((PRD11, PRD12, PRD13, PRD15, TST03), (PRD14, TST04)*) , TYPE=CNC	
CHPID PATH=(2F), SHARED,	*
PARTITION=((PRD03, PRD06, PRD07, PRD09, TST01, TST02), (PRD04, * PRD05, PRD08, PRD10)), TYPE=CNC	
CHPID PATH=(32), SHARED,	*
PARTITION=((PRD05, PRD06, PRD08, PRD10, TST01, TST02), (PRD03, * PRD04, PRD07, PRD09)), TYPE=CNC	
CHPID PATH=(4F), SHARED,	*
PARTITION=(PRD01, PRD02), TYPE=CNC	
CHPID PATH=(58), SHARED,	*
PARTITION=((PRD11, PRD12, PRD14, PRD15, TST04), (PRD13, TST03)*) , TYPE=CNC	
CHPID PATH=(85), SHARED,	*
PARTITION=((PRD03, PRD04, PRD05, PRD09, PRD10), (PRD06, PRD07, * PRD08, TST01, TST02)), TYPE=CNC	
CHPID PATH=(D4), SHARED,	*
PARTITION=(PRD01, PRD02), TYPE=CNC	
CHPID PATH=(E3), SHARED,	*
PARTITION=(PRD02, PRD01), TYPE=CNC	

Figure 30. IOCP Example for VTSS01

Appendix D. VSM4 Configuration

The newest generation VTSS is the VSM4, which provides the following advantages over its predecessors:

- Enhanced connectivity options.
- Greater throughput.
- Greater VTSS capacity.
- 4x the number of VTDs and 3x the maximum number of VTVs per VTSS.
- Improved reliability and serviceability.

Table 34 summarizes the VSM3 to VSM4 enhancements that you see from a software and system configuration perspective.

Table 34. VSM3 to VSM4 Comparison: Software and System Configuration Enhancements

Product Feature	VSM3	VSM4
ESCON Interfaces	<p>16 total where:</p> <ul style="list-style-type: none"> • 2 to 14 can be host channels • 2 to 8 can be Nearlink/CLINK connections 	<p>32 total where:</p> <ul style="list-style-type: none"> • 2 to 28 can be host channels • 2 to 16 can be Nearlink/CLINK connections <p>Note: VSM4s are shipped with 16 ports enabled. With the 16 ports enabled option, only the top port on each CIP is enabled (Port 0 or Port 2). 32 ports enabled is an optional, separately priced feature that is activated via microcode diskette.</p> <p>On a VSM4 with 32 ports enabled, each ICE3 ESCON interface card contains two pairs of ESCON ports. Each pair is controlled by its own Channel Interface Processor (CIP). Each CIP switches between the two ports, so that only one port can transfer data at a time.</p>



Table 34. VSM3 to VSM4 Comparison: Software and System Configuration Enhancements

Product Feature	VSM3	VSM4
Maximum Logical Paths	128	<p>16 per port for the 16 port standard configuration = 256 logical paths</p> <p>16 per port for the 32 port optional configuration= 512 logical paths</p> <p>Note: VSM4 provides a theoretical maximum of 512 logical paths per VTSS, but you cannot allocate all 512 logical paths for host-to-VTSS connections.</p>
VTDS per VTSS	64	256
Maximum resident VTVs per VTSS	100,000	300,000

VSM4 with 32 Ports

For the 32 port option, the 8 ICE3 cards have four ESCON ports per card as shown in Figure 31.

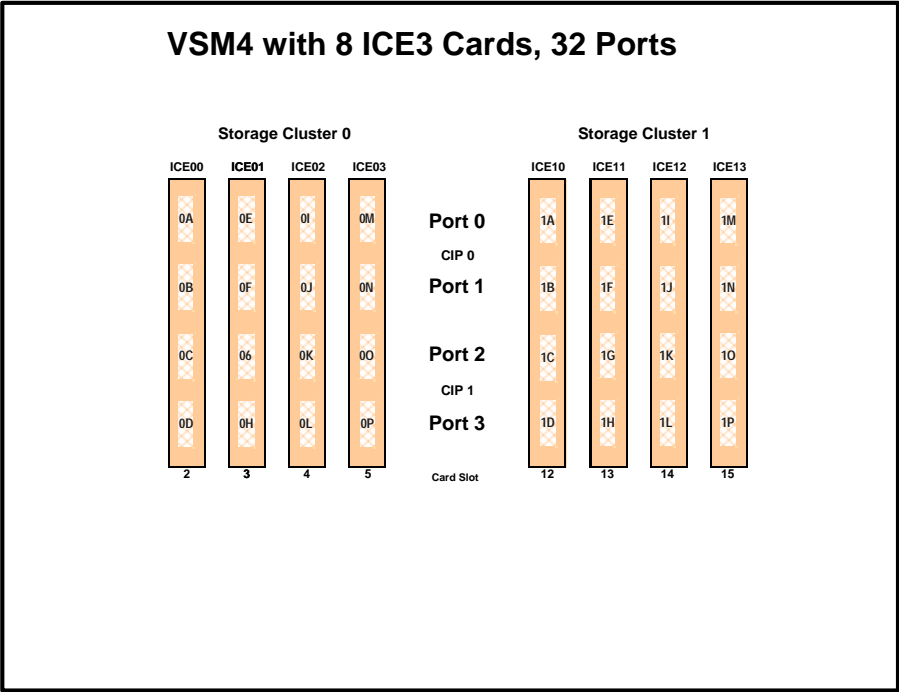


Figure 31. VSM4 with 32 Ports



Note: In Figure 31 and all the other figures in this appendix, the ports are shown with their channel interface identifiers for **enabled** ports (32 in Figure 31). These channel interface identifiers are the values that are required for the CHANIF values that you code for the CONFIG utility. Each value is two characters in length and has a value from 0A to 1P. The first digit is the VTSS cluster ID (valid values are 0 or 1). The second digit is the group or adapter ID (valid values are A to P).

In Figure 31 on page 127, note the following:

- Each ICE3 card contains two pairs of ESCON ports. Each pair is controlled by its own Channel Interface Processor (CIP). Each CIP switches between the two ports, so that **only one port** can transfer data at a time.
- For a VSM4, each CIP can operate with only *one* of two “personalities”, which is set at the VTSS LOP:
 - *Host Mode*. In Host Mode, either or both ESCON ports can connect to host CPU channels, including via ESCON Director(s) or channel extenders. Ports of a CIP in Host mode **cannot** connect to RTDs or to Secondary VTSSs via CLINKS. Note, however, that Secondary VTSSs must have an ESCON port in Host Mode to connect via a CLINK *from* an ESCON port in Nearlink Mode in a Primary VTSS.
 - *Nearlink Mode*. In Nearlink Mode, either or both ESCON ports can connect to an RTD or via a CLINK to a Secondary VTSS. Ports of a CIP in Nearlink mode **cannot** connect to host CPU channels. You can set a **maximum** of 8 CIPs to Nearlink Mode, and here's the important fine print: only **one** Nearlink port per CIP is active at one time. What are Best Practices for optimizing port operations? See Table 35...

Table 35. Optimizing VSM4 Port Operations

Configuration - Two Ports on a CIP	Best Practices
Two CLINKs	Don't usebecause only one port can be active at a time. If you're doing Clustered VTSS, you want all CLINK connections to be active all the time.
CLINK and RTD	An advantage in Degraded Cluster Mode. You normally have fewer RTDs on the Primary VTSS because the Secondary is doing most of the migrations. If you have an offline RTD on the same CIP as an active CLINK, if the Secondary fails you can vary the CLINK offline to VTCS and bring the RTD online to handle more workload on the Primary.

Table 35. Optimizing VSM4 Port Operations

Configuration - Two Ports on a CIP	Best Practices
Two RTDs	<p>An advantage for the following:</p> <ul style="list-style-type: none"> • Optimize use of local and remote RTDs. During busy shifts, use only the local RTD on the CIP. During quiet periods, switch to the remote RTD for deep archive and DR work. • Optimize use of different drive technologies. As described in the previous bullet, use a T9840 as a local RTD, then switch to a T9940 for deep archive. You can also use this feature to migrate from older drive technology (such as 9490) to newer technology (such as 9840). Use Management and Storage Classes to read in data from older media, then switch to the newer technology drive to place data on new media. This technique effectively gives you greater physical connectivity to different drive technologies without incurring the overhead of full time, real time ESCON connections to each drive type. <p>Note that Because of the “only one active” rule, if an RTD on one port is migrating or recalling a VTV, the RTD on the second port cannot be accessed until the operation on the first port completes (the RTD on the second port is in “suspend” mode, as shown by the D RTD command/utility). Best Practices suggests, therefore, that RTDs that must be active simultaneously should connect to different CIPs. One more piece of fine print: If you have two RTDs on a CIP, you can't share them between VTSSs.</p>

- On a VSM4 with 32 ports enabled, you have a **theoretical** maximum of 512 logical paths on the VSM4. However, you must have *some* RTD connections so you cannot allocate *all* 512 logical paths for host-to-VTSS connections. What's the **minimum** number of RTDs? Well, it's like this: (1) CONFIG **will not allow** fewer than 2 RTDS per VTSS. (2) CONFIG cannot check device type, but StorageTek **strongly recommends** at least two RTDS of each device type in each ACS to which the VTSS is attached....otherwise, you can seriously compromise error recovery, and also impact the efficiency of space reclamation. If you had only two RTDs, Best Practices would suggest that you connect them to different ICE3 cards...and once you've done that, you've effectively used up 4 Nearlink ports due to the "CIP personality" nature of the ICE3 card. Therefore, in an 8 ICE card configuration, this leaves 28 available ports for host-to-VTSS ESCON channel connections, which equals a **maximum** of 16 x 28 or 448 logical paths. For more information, see "Logical Paths for VSM 4 with 32 Ports" on page 146.
- A **host** logical path is the communication path between a host and all of the 256 VTDs within the VSM4. Table 36 summarizes the configuration options and maximum host logical paths for a VSM4 with 32 enabled ports.

Table 36. VSM\$8 Configuration Options - 32 Ports

Host CIPs	Maximum Host Connections	Nearlink CIPs	Max Nearlink Connections	Maximum Host Logical Paths
8	16	8	16	256
9	18	7	14	288
10	20	6	12	320
11	22	5	10	352
12	24	4	8	384
14	28	2	4	448

- In HCD:
 - From a single MVS host, you can only define 8 channels (CHPIDs) running to a single control unit (single VSM4). Best Practices would suggest that you make all the connections from a single host to different CIPs.
 - You use the CNTLUNIT statement to define each VSM4 as 16 3490 images.
 - You use the IODEVICE statement to define the 16 VTDs that are associated with each 3490 image.

VSM4 Configuration Examples - 32 Ports

For VSM4s with 32 ports, let's look at two examples of port configurations:

- “VSM4 Configuration Example: 16 Host Ports, 16 RTD Ports” on page 132
- “VSM 4 Configuration Example: 20 Host Ports, 12 RTD Ports” on page 134

For a VSM4 host gen example, see “IOCP Example for Single MVS Host Connected to a VSM4 Via ESCON Directors” on page 144.

VSM4 Configuration
Example: 16 Host
Ports, 16 RTD Ports

Figure 32 shows CONFIG channel interface identifiers of 16 for hosts, 16 for RTDs for a VSM4.

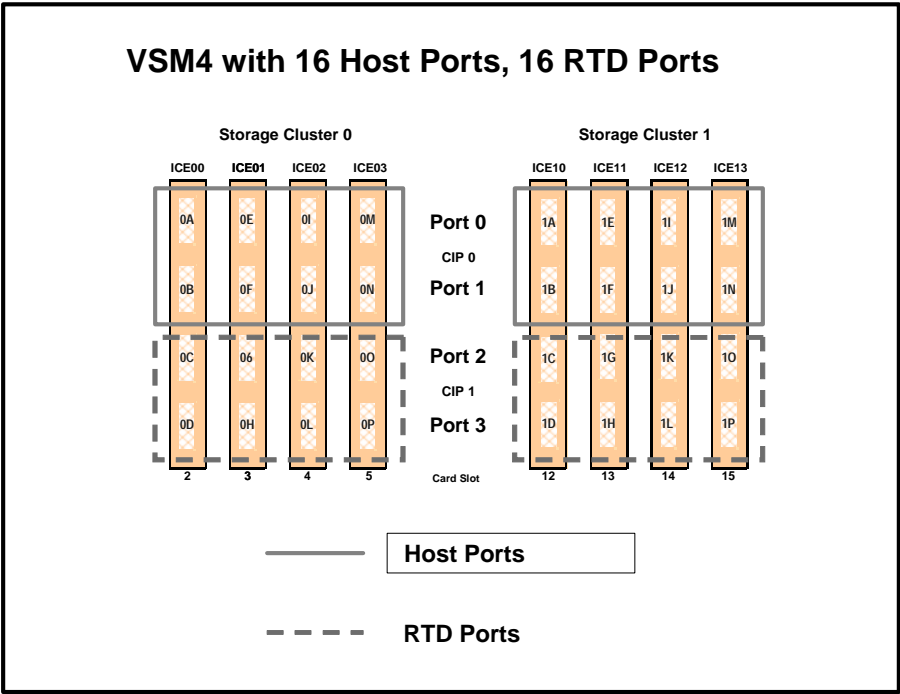


Figure 32. VSM4 with 16 Host Ports, 16 RTD Ports

CONFIG Example for
VSM4 with 16 Host
Ports, 16 RTD Ports

Figure 33 shows example CONFIG JCL to define the VSM4 configuration shown in Figure 32 on page 132.

```
//CREATECFG EXEC PGM=SWSADMIN,PARM='MIXED'
//STEPLIB DD DSN=h1q.SLSLINK,DISP=SHR
//SLSCNTL DD DSN=FEDB.VSMLMULT.DBASEPRM,DISP=SHR
//SLSCNTL2 DD DSN=FEDB.VSMLMULT.DBASESEC,DISP=SHR
//SLSSTBY DD DSN=FEDB.VSMLMULT.DBASESTBY,DISP=SHR
//SLSPRINT DD SYSOUT=*
//SLSIN DD *
CONFIG
GLOBAL MAXVTV=32000 MVCFREE=40
RECLAIM THRESHLD=70 MAXMVC=40 START=35
VTVVOL LOW=905000 HIGH=999999 SCRATCH
VTVVOL LOW=C00000 HIGH=C25000 SCRATCH
VTVVOL LOW=RMM000 HIGH=RMM020 SCRATCH
MVCVOL LOW=N25980 HIGH=N25989
MVCVOL LOW=N35000 HIGH=N35999
VTSS NAME=VSM401 LOW=70 HIGH=80 MAXMIG=8 RETAIN=5
RTD NAME=VSM42A00 DEVNO=2A00 CHANIF=0C
RTD NAME=VSM42A01 DEVNO=2A01 CHANIF=0D
RTD NAME=VSM42A02 DEVNO=2A02 CHANIF=0G
RTD NAME=VSM42A03 DEVNO=2A03 CHANIF=0H
RTD NAME=VSM42A04 DEVNO=2A04 CHANIF=0K
RTD NAME=VSM42A05 DEVNO=2A05 CHANIF=0L
RTD NAME=VSM42A06 DEVNO=2A06 CHANIF=0O
RTD NAME=VSM42A07 DEVNO=2A07 CHANIF=0P
RTD NAME=VSM42A08 DEVNO=2A08 CHANIF=1C
RTD NAME=VSM42A09 DEVNO=2A09 CHANIF=1D
RTD NAME=VSM42A0A DEVNO=2A0A CHANIF=1G
RTD NAME=VSM42A0B DEVNO=2A0B CHANIF=1H
RTD NAME=VSM42A0C DEVNO=2A0C CHANIF=1K
RTD NAME=VSM42A0D DEVNO=2A0D CHANIF=1L
RTD NAME=VSM42A0E DEVNO=2A0E CHANIF=1O
RTD NAME=VSM42A0F DEVNO=2A0F CHANIF=1P
VTD LOW=9900 HIGH=99FF
```

Figure 33. CONFIG example: VSM4 with 16 Host Ports, 16 RTD Ports

VSM 4 Configuration
Example: 20 Host
Ports, 12 RTD Ports

Figure 34 shows port assignments of 20 for hosts, 12 for RTDs for a VSM4.

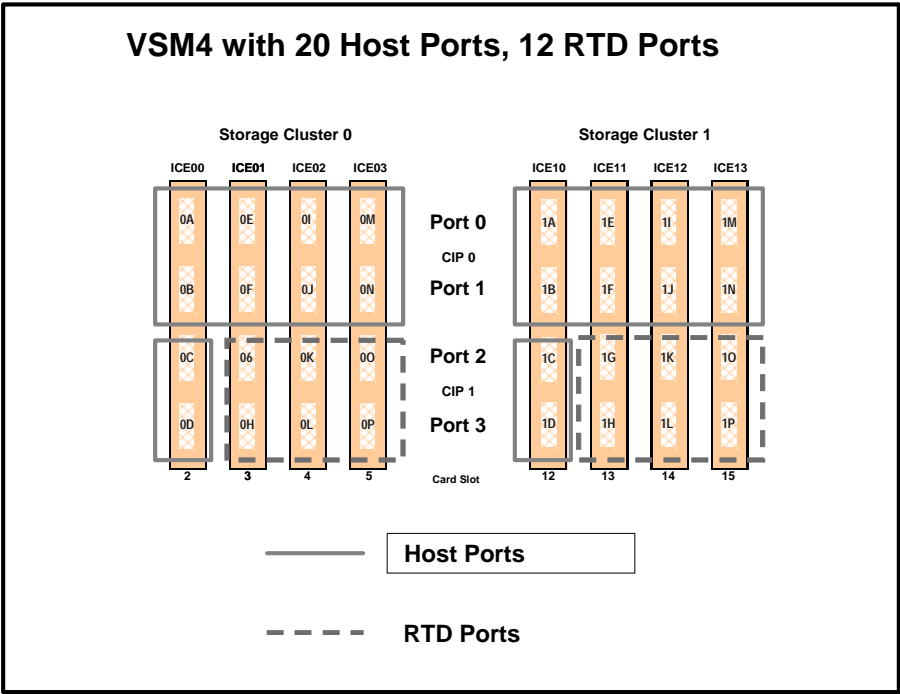


Figure 34. VSM4 with 20 Host Ports, 12 RTD Ports

CONFIG Example for
VSM4 with 20 Host
Ports, 12 RTD Ports

Figure 35 shows example CONFIG JCL to define the VSM4 configuration shown in Figure 34 on page 134.

```
//CREATECFG EXEC PGM=SWSADMIN,PARM='MIXED'
//STEPLIB DD DSN=h1q.SLSLINK,DISP=SHR
//SLSCNTL DD DSN=FEDB.VSMLMULT.DBASEPRM,DISP=SHR
//SLSCNTL2 DD DSN=FEDB.VSMLMULT.DBASESEC,DISP=SHR
//SLSSTBY DD DSN=FEDB.VSMLMULT.DBASETBY,DISP=SHR
//SLSPRINT DD SYSOUT=*
//SLSIN DD *

CONFIG
GLOBAL MAXVTV=32000 MVCFREE=40
RECLAIM THRESHLD=70 MAXMVC=40 START=35
VTVOL LOW=905000 HIGH=999999 SCRATCH
VTVOL LOW=C00000 HIGH=C25000 SCRATCH
VTVOL LOW=RMM000 HIGH=RMM020 SCRATCH
MVCVOL LOW=N25980 HIGH=N25989
MVCVOL LOW=N35000 HIGH=N35999
VTSS NAME=VSM401 LOW=70 HIGH=80 MAXMIG=6 RETAIN=5
RTD NAME=VSM42A00 DEVNO=2A00 CHANIF=0G
RTD NAME=VSM42A01 DEVNO=2A01 CHANIF=0H
RTD NAME=VSM42A02 DEVNO=2A02 CHANIF=0K
RTD NAME=VSM42A03 DEVNO=2A03 CHANIF=0L
RTD NAME=VSM42A04 DEVNO=2A04 CHANIF=0O
RTD NAME=VSM42A05 DEVNO=2A05 CHANIF=0P
RTD NAME=VSM42A06 DEVNO=2A06 CHANIF=1G
RTD NAME=VSM42A07 DEVNO=2A07 CHANIF=1H
RTD NAME=VSM42A08 DEVNO=2A08 CHANIF=1K
RTD NAME=VSM42A09 DEVNO=2A09 CHANIF=1L
RTD NAME=VSM42A0A DEVNO=2A0A CHANIF=1O
RTD NAME=VSM42A0B DEVNO=2A0B CHANIF=1P
VTD LOW=9900 HIGH=99FF
```

Figure 35. CONFIG *example: VSM4 with 20 Host Ports, 12 RTD Ports*

VSM4 with 16 Ports

For the 16 port option, the 8 ICE3 cards have two ESCON ports per card as shown in Figure 36.

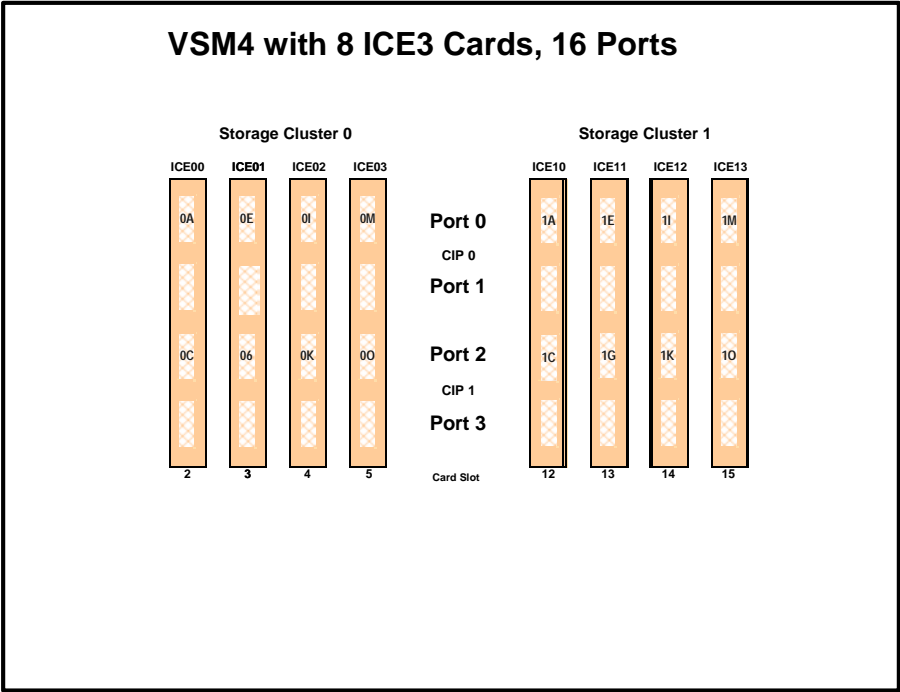


Figure 36. VSM4 with 16 Ports



Note: In Figure 36 and all the other figures in this appendix, the ports are shown with their channel interface identifiers for **enabled** ports (16 in Figure 36). These channel interface identifiers are the values that are required for the CHANIF values that you code for the CONFIG utility. Each value is two characters in length and has a value from 0A to 1P. The first digit is the VTSS cluster ID (valid values are 0 or 1). The second digit is the group or adapter ID (valid values are A to P).

In Figure 31 on page 127, note the following:

- Each ICE3 card has two CIPs with a **single port enabled** on each CIP. As with the 32 port option, each CIP can operate with only *one* of two “personalities”, which is set at the VTSS LOP:
 - *Host Mode*. In Host Mode, the single ESCON port can connect to host CPU channels, including via ESCON Director(s) or channel extenders. Ports of a CIP in Host mode **cannot** connect to RTDs or to Secondary VTSSs via CLINKS. Note, however, that Secondary VTSSs must have an ESCON port in Host Mode to connect via a CLINK *from* an ESCON port in Nearlink Mode in a Primary VTSS.
 - *Nearlink Mode*. In Nearlink Mode, the single ESCON port can connect to an RTD or via a CLINK to a Secondary VTSS. Ports of a CIP in Nearlink mode **cannot** connect to host CPU channels.

You can set a **maximum** of 8 CIPs to Nearlink Mode. Therefore, in a 16 port configuration, the single port on a CIP can be either a CLINK or an RTD connection.

- On a VSM4 with 16 ports enabled, you have a **theoretical** maximum of 256 logical paths on the VSM4. However, you must have *some* RTD connections so you cannot allocate *all* 256 logical paths for host-to-VTSS connections. What’s the **minimum** number of RTDs? Well, it’s like this: (1) CONFIG will not allow fewer than 2 RTDS per VTSS. (2) CONFIG cannot check device type, but StorageTek **strongly recommends** at least two RTDS of each device type in each ACS to which the VTSS is attached....otherwise, you can seriously compromise error recovery and also impact the efficiency of space reclamation. If you had only two RTDs, Best Practices would suggest that you connect them to different ICE3 cards. ...and once you’ve done that, you’ve effectively used up 4 Nearlink ports. Therefore, in an 8 ICE card 16 port configuration, this leaves 12 available ports for host-to-VTSS ESCON channel connections, which equals a **maximum** of 16 x 12 or 192 logical paths. For more information, see “Logical Paths for VSM 4 with 32 Ports” on page 146.
- A **host** logical path is the communication path between a host and all of the 256 VTDs within the VSM4. Table 36 summarizes the configuration options and maximum host logical paths for a VSM4 with 16 enabled ports.

Table 37. VSM4 Configuration Options - 16 Ports

Host CIPs	Maximum Host Connections	Nearlink CIPs	Max Nearlink Connections	Maximum Host Logical Paths
8	8	8	8	128
9	9	7	7	144
10	10	6	6	160
11	11	5	5	176

Table 37. VSM4 Configuration Options - 16 Ports

Host CIPs	Maximum Host Connections	Nearlink CIPs	Max Nearlink Connections	Maximum Host Logical Paths
12	12	4	4	192
14	14	2	2	224

- In HCD:
 - From a single MVS host, you can only define 8 channels (CHPIDs) running to a single control unit (single VSM4). Best Practices would suggest that you make all the connections from a single host to different CIPs.
 - You use the CNTLUNIT statement to define each VSM4 as 16 3490 images.
 - You use the IODEVICE statement to define the 16 VTDs that are associated with each 3490 image.

VSM4 Configuration Examples - 16 Ports

For VSM4s with 16 ports, let's look at two examples of port configurations:

- “VSM4 Configuration Example: 8 Host Ports, 8 RTD Ports” on page 140
- “VSM 4 Configuration Example: 10 Host Ports, 6 RTD Ports” on page 142

For a VSM4 host gen example, see “IOCP Example for Single MVS Host Connected to a VSM4 Via ESCON Directors” on page 144.

VSM4 Configuration
Example: 8 Host
Ports, 8 RTD Ports

Figure 37 shows CONFIG channel interface identifiers of 8 for hosts, 8 for RTDs for an 8 ICE3 card VSM4 with 16 ports.

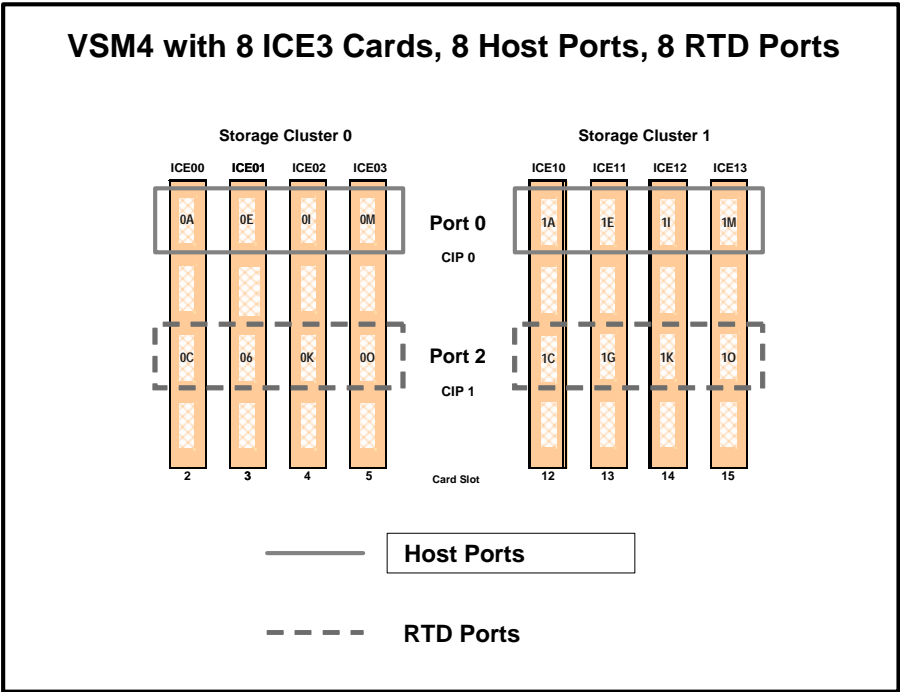


Figure 37. VSM4 with 8 Host Ports, 8 RTD Ports

CONFIG Example for
VSM4 with 8 Host
Ports, 8 RTD Ports

Figure 38 shows example CONFIG JCL to define the VSM4 configuration shown in Figure 37 on page 140.

```
//CREATECFG EXEC PGM=SWSADMIN,PARM='MIXED'
//STEPLIB DD DSN=h1q.SLSLINK,DISP=SHR
//SLSCNTL DD DSN=FEDB.VSMLMULT.DBASEPRM,DISP=SHR
//SLSCNTL2 DD DSN=FEDB.VSMLMULT.DBASESEC,DISP=SHR
//SLSSTBY DD DSN=FEDB.VSMLMULT.DBASEBY,DISP=SHR
//SLSPRINT DD SYSOUT=*
//SLSIN DD *
CONFIG
GLOBAL MAXVTV=32000 MVCFREE=40
RECLAIM THRESHLD=70 MAXMVC=40 START=35
VTVVOL LOW=905000 HIGH=999999 SCRATCH
VTVVOL LOW=C00000 HIGH=C25000 SCRATCH
VTVVOL LOW=RMM000 HIGH=RMM020 SCRATCH
MVCVOL LOW=N25980 HIGH=N25989
MVCVOL LOW=N35000 HIGH=N35999
VTSS NAME=VSM401 LOW=70 HIGH=80 MAXMIG=8 RETAIN=5
RTD NAME=VSM42A00 DEVNO=2A00 CHANIF=0C
RTD NAME=VSM42A02 DEVNO=2A02 CHANIF=0G
RTD NAME=VSM42A04 DEVNO=2A04 CHANIF=0K
RTD NAME=VSM42A06 DEVNO=2A06 CHANIF=0O
RTD NAME=VSM42A08 DEVNO=2A08 CHANIF=1C
RTD NAME=VSM42A0A DEVNO=2A0A CHANIF=1G
RTD NAME=VSM42A0C DEVNO=2A0C CHANIF=1K
RTD NAME=VSM42A0E DEVNO=2A0E CHANIF=1O
VTD LOW=9900 HIGH=99FF
```

Figure 38. CONFIG *example: VSM4 with 8 Host Ports, 8 RTD Ports*

VSM 4 Configuration
Example: 10 Host
Ports, 6 RTD Ports

Figure 39 shows port assignments of 10 for hosts, 6 for RTDs for a VSM4 with 16 ports.

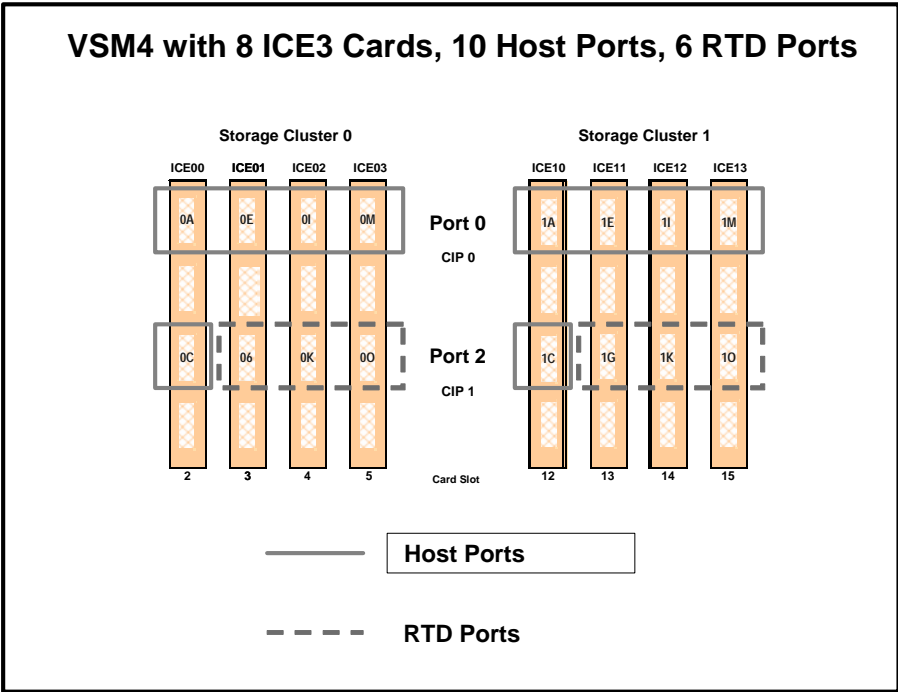


Figure 39. VSM4 with 10 Host Ports, 6 RTD Ports

CONFIG Example for
VSM4 with 10 Host
Ports, 6 RTD Ports

Figure 40 shows example CONFIG JCL to define the VSM4 configuration shown in Figure 39 on page 142.

```
//CREATECFG EXEC PGM=SWSADMIN,PARM='MIXED'
//STEPLIB DD DSN=h1q.SLSLINK,DISP=SHR
//SLSCNTL DD DSN=FEDB.VSMLMULT.DBASEPRM,DISP=SHR
//SLSCNTL2 DD DSN=FEDB.VSMLMULT.DBASESEC,DISP=SHR
//SLSSTBY DD DSN=FEDB.VSMLMULT.DBASEBY,DISP=SHR
//SLSPRINT DD SYSOUT=*
//SLSIN DD *
CONFIG
GLOBAL MAXVTV=32000 MVCFREE=40
RECLAIM THRESHLD=70 MAXMVC=40 START=35
VTVVOL LOW=905000 HIGH=999999 SCRATCH
VTVVOL LOW=C00000 HIGH=C25000 SCRATCH
VTVVOL LOW=RMM000 HIGH=RMM020 SCRATCH
MVCVOL LOW=N25980 HIGH=N25989
MVCVOL LOW=N35000 HIGH=N35999
VTSS NAME=VSM401 LOW=70 HIGH=80 MAXMIG=8 RETAIN=5
RTD NAME=VSM42A02 DEVNO=2A02 CHANIF=0G
RTD NAME=VSM42A04 DEVNO=2A04 CHANIF=0K
RTD NAME=VSM42A06 DEVNO=2A06 CHANIF=0O
RTD NAME=VSM42A0A DEVNO=2A0A CHANIF=1G
RTD NAME=VSM42A0C DEVNO=2A0C CHANIF=1K
RTD NAME=VSM42A0E DEVNO=2A0E CHANIF=1O
VTD LOW=9900 HIGH=99FF
```

Figure 40. CONFIG *example: VSM4 with 10 Host Ports, 6 RTD Ports*

IOCP Example for Single MVS Host Connected to a VSM4 Via ESCON Directors

Figure 41 shows a configuration diagram for a single MVS host connected to a VSM4 via ESCON Directors, and Figure 42 on page 145 shows example IOCP statements for this configuration. **Note that:**

- From MVSA, you define 8 CHPIDs, with each path switched in the ESCON Director, for a total of 8 channels running to the VSM4.
- You code 16 CNTLUNIT statements to define the VSM4 as 16 3490 images.
- You code IODEVICE statement to define the 16 VTDs that are associated with each 3490 image.

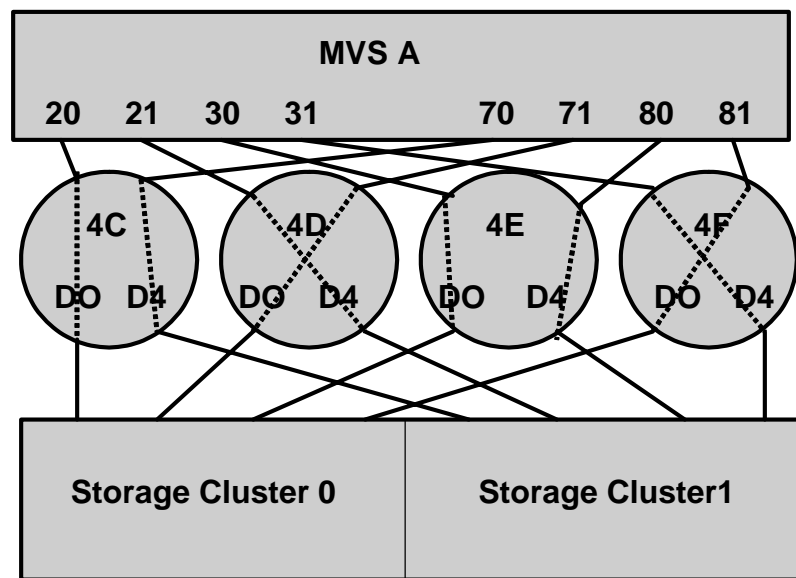


Figure 41. Configuration Diagram: Single MVS Host Connected to a VSM4 via ESCON Directors

```

ESCD4C  CHPID PATH=(20,70),TYPE=CNC,SWITCH=4C
ESCD4D  CHPID PATH=(21,71),TYPE=CNC,SWITCH=4D
ESCD4E  CHPID PATH=(30,80),TYPE=CNC,SWITCH=4E
ESCD4F  CHPID PATH=(31,81),TYPE=CNC,SWITCH=4F

CU1      CNTLUNIT CUNUMBR=001,
          PATH=(20,21,30,31,70,71,80,81),
          LINK=(D0,D4,D0,D4,D4,D0,D4,D0),
          UNIT=3490,CUADD=0,
          UNITADD=(00,16))

STRING1  IODEVICE ADDRESS=(0500,16),
          CUNUMBR=(001),
          UNIT=3490,
          UNITADD=00,STADET=Y

CU2      CNTLUNIT CUNUMBR=002,
          PATH=(20,21,30,31,70,71,80,81),
          LINK=(D0,D4,D0,D4,D4,D0,D4,D0),
          UNIT=3490,CUADD=1,
          UNITADD=(00,16))

STRING2  IODEVICE ADDRESS=(0510,16),
          CUNUMBR=(002),
          UNIT=3490,
          UNITADD=00,STADET=Y
          .
          .
          .

CU15     CNTLUNIT CUNUMBR=015,
          PATH=(20,21,30,31,70,71,80,81),
          LINK=(D0,D4,D0,D4,D4,D0,D4,D0),
          UNIT=3490,CUADD=E,
          UNITADD=(00,16))

STRING15 IODEVICE ADDRESS=(05E0,16),
          CUNUMBR=(015),
          UNIT=3490,
          UNITADD=00,STADET=Y

CU16     CNTLUNIT CUNUMBR=016,
          PATH=(20,21,30,31,70,71,80,81),
          LINK=(D0,D4,D0,D4,D4,D0,D4,D0),
          UNIT=3490,CUADD=F,
          UNITADD=(00,16))

STRING16 IODEVICE ADDRESS=(05F0,16),
          CUNUMBR=(016),
          UNIT=3490,
          UNITADD=00,STADET=Y

```

Figure 42. IOCP Example: Single MVS Host Connected to a VSM4 via ESCON Directors

Logical Paths for VSM 4 with 32 Ports

A VSM4 with 32 ports has 4x the number of logical paths available to VSM2s and VSM3s. Does this mean that a VSM4 has enough logical paths for connectivity, redundancy, and throughput for *all* attached hosts? Even with 16 RTDs and 31 hosts attached, the answer is “yes” as shown in Figure 43.

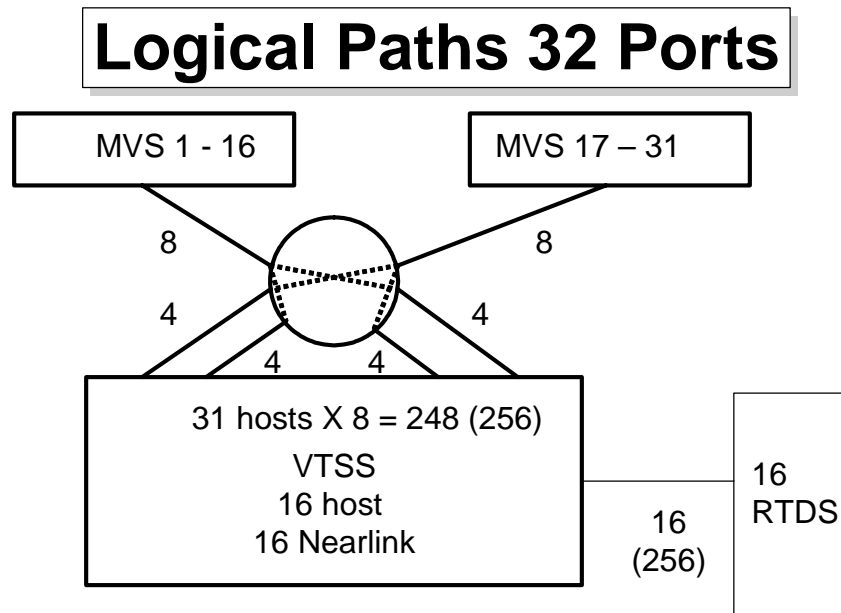


Figure 43. Logical Paths for VSM 4 with 32 Ports, 31 Hosts, 16 RTDs

In Figure 43:

- The 16 RTDs consume 16×16 or 256 logical paths.
- Looking back at “VSM2 and VSM3 Logical Path Planning and Configuration Example” on page 107, the **maximum** logical paths we allocated for a VSM2/3 was 4 to a host requiring maximum throughput (which also satisfied the redundancy/connectivity requirements). Therefore, if we allocated **double** that number, or 8 logical paths, for each of the 31 hosts in this configuration, we only consume 248, or 8 less than the logical paths remaining for host connections.

Therefore, logical path allocation isn’t an issue, as it was with VSM2s and VSM3s.

Glossary

A

access method A technique for moving data between processor storage and input/output devices.

ACS *See* Automated Cartridge System.

ACSid A method used to identify an ACS. An ACSid is the result of defining the SLIALIST macro during the library generation (LIBGEN) process. The first ACS listed in this macro acquires a hexadecimal identifier of 00, the second ACS listed acquires a hexadecimal identifier of 01, and so forth, until all ACSs are identified.

ACS routine An SMS term, referring to automatic class selection routine. Not to be confused with the HSC term, ACS, referring to automatic cartridge system.

AMT automatic migration threshold.

APF Authorized Program Facility.

APPL VTAM APPLID definition for the HSC.

archiving The storage of backup files and associated journals, usually for a given period of time.

audit A VSM audit (which is not the same as an HSC audit) reconstructs VTV and MVC information.

Automated Cartridge System (ACS) The library subsystem consisting of one or two LMUs, and from 1 to 16 attached LSMs.

automated library *See* library.

automatic mode A relationship between an LSM and all attached hosts. LSMs operating in automatic mode handle cartridges without operator intervention. This is the normal operating mode of an LSM that has been modified online.

automatic migration Migrating VTVs to MVCs that is automatically initiated and controlled by VSM.

automatic migration threshold (AMT) AMT values are percentage values that determine when

virtual tape volume migration begins and ends. VTV migration begins when the VTSS buffer reaches the high AMT and ends when the buffer reaches or falls below the low AMT. These thresholds apply to all VTSSs.

automatic recall Recalling VTVs to the VTSS that is automatically initiated and controlled by VSM.

automatic reclaim Reclaiming MVC space that is automatically initiated and controlled by VSM.

B

block A collection of contiguous records recorded as a unit. Blocks are separated by interblock gaps, and each block may contain one or more records.

buffer A routine or storage used to compensate for a difference in rate of data flow, or time of occurrence of events, when transferring data from one device to another.

C

CA-1 (TMS) Computer Associates Tape Management System. Third-party software by Computer Associates International, Inc.

CAP *See* Cartridge Access Port.

capacity *See* media capacity.

CAPid A CAPid uniquely defines the location of a CAP by the LSM on which it resides. A CAPid is of the form AAL:CC where AA is the ACSid, L is the LSM number, and CC is the CAP number. Some commands and utilities permit an abbreviated CAPid format of AAL.

cartridge The plastic housing around the tape. It is approximately 4 inches (100 mm) by 5 inches (125 mm) by 1 inch (25 mm). The tape is threaded automatically when loaded in a transport. A plastic leader block is attached to the tape for automatic threading. The spine of the cartridge contains a Tri-Optic label listing the VOLSER (tape volume identifier).

Cartridge Access Port (CAP) An assembly which allows an operator to enter/eject cartridges during automated operations. The CAP is located on the access door of an LSM. (*see also*, standard CAP, enhanced CAP, WolfCreek CAP, WolfCreek optional CAP.)

Cartridge Scratch Loader An optional feature for the Cartridge Drive. It allows the automatic loading of premounted tape cartridges or the manual loading of single tape cartridges.

cartridge system tape The basic tape cartridge media that is used with 4480, 4490, or 9490 Cartridge Subsystems. They are visually identified by a one-color cartridge case.

CAW *See* Channel Address Word.

CDRM Cross Domain Resource Manager definition (if not using existing CDRMs).

CDRSC Cross Domain Resource definition.

CDS *See* control data set.

CE Channel End.

cell A storage slot in the LSM that is used to store a tape cartridge.

Central Support Remote Center (CSRC) *See* Remote Diagnostics Center.

CFT Customer field test.

channel A device that connects the host and main storage with the input and output control units.

Channel Address Word (CAW) An area in storage that specifies the location in main storage at which a channel program begins.

channel command A command received by a CU from a channel.

Channel Status Word (CSW) An area in storage that provides information about the termination of input/output operations.

check Detection of an error condition.

CI Converter/Interpreter (JES3).

Clink (cluster link). The path between a primary VTSS and secondary VTSS in a cluster. The Clink path is used to copy replicate VTVs from the primary to the secondary.

Cluster. Two VTSSs which are physically cabled together by Clink paths and are defined in CONFIG as a cluster. A cluster consists of a primary and a secondary VTSS. VTVs with the replicate attribute attached will be copied from the primary to the secondary as soon as possible after dismount time.

connected mode A relationship between a host and an ACS. In this mode, the host and an ACS are capable of communicating (at least one station to this ACS is online).

control data set (CDS) The HSC database. In addition to the current information in the CDS, VSM keeps all its persistent data in the CDS as well.

control data set allocation map A CDS subfile that marks individual blocks as used or free.

control data set data blocks CDS blocks that contain information about the library and its configuration or environment.

control data set directory A part of the CDS that maps its subdivision into subfiles.

control data set pointer blocks CDS blocks that contain pointers to map data blocks belonging to a subfile.

control data set recovery area A portion of the CDS reserved for maintaining integrity for updates that affect multiple CDS blocks.

control data set subfile A portion of the CDS consisting of Data Blocks and Pointer Blocks containing related information.

Control Unit (CU) A microprocessor-based unit situated logically between a host channel (or channels) and from two to sixteen tape transports. It functions to translate channel commands into tape transport commands, send transport status to the channel(s), and pass data between the channel(s) and transport(s).

conventional Nearline transport An HSC-controlled transport that is not defined to VSM as an RTD.

cross-host recovery The ability for one host to perform recovery for another host that has failed.

CSE Customer Service Engineer.

CSI Consolidated System Inventory.

CSL Cartridge Scratch Loader.

CSRC Central Support Remote Center (*See* Remote Diagnostics Center)

CSW Channel Status Word.

CU *See* Control Unit.

D

DAE Dump Analysis Elimination.

DASD Direct access storage device.

data Any representations such as characters or analog quantities to which meaning is, or might be, assigned.

data class A collection of allocation and space attributes, defined by the storage administrator, that are used to create a data set.

data compaction An algorithmic data-reduction technique that encodes data from the host and stores it in less space than unencoded data. The original data is recovered by an inverse process call decompaction.

data-compaction ratio The number of host data bytes divided by the number of encoded bytes. It is variable depending on the characteristics of the data being processed. The more random the data stream, the lower the opportunity to achieve compaction.

Data Control Block (DCB) A control block used by access routines in storing and retrieving data.

data set The major unit of data storage and retrieval, consisting of a collection of data in one of several prescribed arrangements and described by control information to which the system has access.

data streaming A continuous stream of data being transmitted in character or binary-digit form, using a specified format.

DBU disk buffer utilization.

DCB Data Control Block.

demand allocation An MVS term meaning that a user has requested a specific unit.

demand migration Migrating VTVs to MVCs that an administrator does with the MIGRATE command or utility.

demand recall Recalling VTVs to the VTSS that an administrator does with the RECALL command or utility.

demand reclaim Reclaiming MVC space that an administrator does with the RECLAIM command or utility.

device number A four-digit hexadecimal number that uniquely identifies a device attached to a processor.

device separation The HSC function which *forces* the MVS device selection process to choose either a nonlibrary transport or a transport in a particular ACS, based on the location of the volume (specific requests) or the given subpool rules in effect (nonspecific request).

DFP Data Facility Product. A program that isolates applications from storage devices, storage management, and storage device hierarchy management.

DFSMS Refers to an environment running MVS/ESA SP and DFSMS/MVS, DFSORT, and RACF. This environment helps automate and centralize the management of storage through a combination of hardware, software, and policies.

DFSMS ACS routine A sequence of instructions for having the system assign data class, storage class, management class, and storage group for a data set.

directed allocation The HSC function of *influencing* MVS's selection of library transports. For a specific request, the HSC influences MVS to choose a transport requiring the fewest number of pass-thrus; for a nonspecific (scratch) request, HSC's influencing is based on the given subpool rules in effect.

disconnected mode A relationship between a host and an ACS. In this mode, the host and an ACS are

not capable of communicating (there are no online stations to this ACS).

disk buffer utilization (DBU). The ratio of used to total VTSS buffer capacity.

DOMed Pertaining to a console message that was previously highlighted during execution, but is now at normal intensity.

drain The deletion of data from an MVC. May be accompanied by a “virtual” eject to prevent the MVC from being reused.

drive loaded A condition of a tape drive in which a tape cartridge has been inserted in the drive, and the tape has been threaded to the beginning-of-tape position.

DSI Dynamic System Interchange (JES3).

dual LMU A hardware/u–software feature that provides a redundant LMU capability.

dual LMU HSC release 1.1.0 or later that automates a switchover to the standby LMU in a dual LMU configuration.

dump To write the contents of storage, or of a part of storage, usually from an internal storage to an external medium, for a specific purpose such as to allow other use of storage, as a safeguard against faults or errors, or in connection with debugging.

Dynamic Device Reconfiguration (DDR) A facility that allows a demountable volume to be moved, and repositioned if necessary, without abnormally terminating the job or repeating the initial program load procedure.

E

Ecart Cartridge system tape with a length of 1100 feet that can be used with 4490 cartridge drives. These tapes are visually identified by a two-tone colored case.

EDL See eligible device list.

eligible device list A group of tape drives that are available to satisfy an allocation request.

enhanced CAP An enhanced CAP contains two forty-cell magazine-style CAPs and a one-cell priority CAP (PCAP). Each forty-cell CAP holds

four removable magazines of ten cells each. An LSM access door with an enhanced CAP contains no cell locations for storing cartridges. An enhanced CAP is ordered as Feature Number CC80. (*see also*, Cartridge Access Port (CAP), standard CAP, WolfCreek CAP, WolfCreek optional CAP.)

Effective Recording Density The number of user bytes per unit of length of the recording medium.

eject The LSM robot places a cartridge in a Cartridge Access Port (CAP) so the operator can remove it from the LSM.

ExPR Expert Performance Reporter.

Expert Performance Reporter Expert Performance Reporter collects performance data and generates reports about StorageTek Nearline ACSs and VTSS status and performance. It has an MVS component and a PC component.

Enhanced Capacity Cartridge System Tape Cartridge system tape with increased capacity that can be used with 4490 and 9490 Cartridge Drives. These tapes are visually identified by a two-tone colored case.

EOT End-of-Tape marker.

EPO Emergency Power Off.

ERDS Error Recording Data Set.

EREP Environmental Recording, Editing, Printing.

ERP Error recovery procedures.

error recovery procedures (ERP) Procedures designed to help isolate and, where possible, to recover from errors in equipment.

ExtendedStore Library One or more LSMs with no cartridge drives (CDs) that are attached by pass-thru ports to other LSMs (with CDs) in an ACS. These LSMs provide archive storage for cartridges containing less active data sets. Cartridges can be entered and ejected directly into and out of this LSM though either a standard CAP or an enhanced CAP.

F

file protected Pertaining to a tape volume from which data can be read only. Data cannot be written on or erased from the tape.

format The arrangement or layout of data on a data medium.

G

GB 1,073,741,824 bytes of storage.

GDG Generation Data Group. An MVS data set naming convention. Sequence numbers are appended to the basic data set name to track the generations created for that data set.

GTF Generalized Trace Facility. An MVS facility used to trace software functions and events.

H

HDA Head/disk assembly.

Host Software Component (HSC) That portion of the Automated Cartridge System which executes on host systems attached to an automated library. This component acts as the interface between the operating system and the rest of the automated library.

host system A data processing system that is used to prepare programs and the operating environments for use on another computer or controller.

HSC Host Software Component.

HSM Hierarchical Storage Manager.

HWS High Watermark Setup. Relates to chains set up for tape transport allocation in JES3.

I

ICRC See Improved Cartridge Recording Capability.

Improved Cartridge Recording Capability (ICRC) An improved data recording mode that, when enabled, can increase the effective cartridge data capacity and the effective data rate when invoked.

ID Identifier or identification.

IDAX Interpreter Dynamic Allocation Exit. This is a subfunction of the DFSMS/MVS subsystem request (SSREQ 55) that the MVS JCL Interpreter and dynamic allocation functions issue for calling

DFSMS ACS routines for management of the data set requested.

IML See Initial Microprogram Load.

index a function performed by the cartridge loader that moves cartridges down the input or output stack one cartridge position. A loader can perform multiple consecutive indexes.

Initial Microprogram Load (IML) A process that activates a machine reset and loads system programs to prepare a computer system for operation. Processors having diagnostic programs activate these programs at IML execution. Devices running u–software reload the functional u–software usually from a floppy diskette at IML execution.

Initial Program Load (IPL) A process that activates a machine reset and loads system programs to prepare a computer system for operation. Processors having diagnostic programs activate these programs at IPL execution. Devices running u–software reload the functional u–software usually from a floppy diskette at IPL execution.

initial value A value assumed until explicitly changed. It must then be explicitly specified in another command to restore the initial value. An initial value for the HSC is the value in effect when the product is installed.

inline diagnostics Diagnostic routines that test subsystem components while operating on a time–sharing basis with the functional u–software in the subsystem component.

input stack The part of the cartridge loader where cartridges are premounted.

intervention required Manual action is needed.

ips Inches per second.

IVP Installation Verification Programs. A package of programs that is run by a user after the library is installed in order to verify that the library is functioning properly.

J

JCL See Job Control Language.

Job Control Language Problem-oriented language designed to express statements in a job that are used to identify the job or describe its requirements to an operating system.

journal The log associated with journaling. The log (stored in a data set) contains a record of completed work and changes to the control data set since the last backup was created.

journaling A technique for recovery that involves creating a backup control data set and maintaining a log of all changes (transactions) to that data set.

K

KB Kilobyte, thousand bytes, or 1024 bytes.

kb kilobit, or thousand bits (10^3 bits).

keyword parameter In command and utility syntax, operands that include keywords and their related values (*see* “positional parameter”). Values are concatenated to the keyword either by an equal sign, “KEYWORD=value,” or by parentheses, “KEYWORD(value).” Keyword parameters can be specified in any order. The HSC accepts (tolerates) multiple occurrences of a keyword. The value assigned to a keyword reflects the last occurrence of a keyword within a command.

L

LAN Local Area Network.

LCU *See* Library Control Unit.

LED *See* Light Emitting Diode.

LIBGEN The process of defining the configuration of the automated library to the host software.

library An installation of one or more ACSs, attached cartridge drives, volumes placed into the ACSs, host software that controls and manages the ACSs and associated volumes, and the library control data set that describes the state of the ACSs.

library control data set *See* control data set.

Library Control Unit (LCU) The portion of the LSM that controls the picking, mounting, dismounting, and replacing of cartridges.

Light Emitting Diode (LED) An electronic device used mainly as an indicator on status panels to show equipment on/off conditions.

LMU Library Management Unit. The portion of the ACS that manages from one to sixteen LSMs and communicates with the host CPU.

loader *See* Cartridge Scratch Loader.

load point The beginning of the recording area on magnetic tape.

Local Area Network (LAN) A computer network in which devices within the network can access each other for data transmission purposes. The LMU and attached LCUs are connected with a local area network.

logical ejection The process of removing a volume from the control data set without physically ejecting it from its LSM location.

LSM Library Storage Module. Provides the storage area for cartridges plus the robot necessary to move the cartridges. The term LSM often means the LCU and LSM combined.

LSMid An LSMid is composed of the ACSid concatenated with the LSM number.

LSM number A method used to identify an LSM. An LSM number is the result of defining the SLIACS macro LSM parameter during a LIBGEN. The first LSM listed in this parameter acquires the LSM number of 0 (hexadecimal), the second LSM listed acquires a hexadecimal number of 1, and so forth, until all LSMs are identified (maximum of sixteen or hexadecimal F).

M

machine initiated maintenance *See* ServiceTek.

magnetic recording A technique of storing data by selectively magnetizing portions of a magnetizable material.

magnetic tape A tape with a magnetizable surface layer on which data can be stored by magnetic recording.

magnetic tape drive A mechanism for moving magnetic tape and controlling its movement.

maintenance facility Hardware contained in the CU and LMU that allows a CSE and the RDC to run diagnostics, retrieve status, and communicate with respective units through their control panels.

management class A collection of management attributes, assigned by the storage administrator, that are used to control the allocation and use of space by a data set. Note that SMS Management Classes are different from VSM Management Classes.

manual mode A relationship between an LSM and all attached hosts. LSMs operating in manual mode have been modified offline and require human assistance to perform cartridge operations.

master LMU The LMU currently controlling the functional work of the ACS in a dual LMU configuration.

MDS Main Device Scheduler (JES3).

media capacity The amount of data that can be contained on storage media and expressed in bytes of data.

micro–software See *v–software* under Symbols.

migration The movement of VTVs from the VTSS to the RTD where the VTVs are stacked onto MVCs. See *automatic migration* and *demand migration*.

MIM Multi–Image Manager. Third–party software by CA Corporation.

mixed configurations Installations containing cartridge drives under ACS control and cartridge drives outside of library control. These configurations cause the Host Software Component to alter allocation to one or the other.

modem Modulator/demodulator. An electronic device that converts computer digital data to analog data for transmission over a telecommunications line (telephone line). At the receiving end, the modem performs the inverse function.

monitor A device that observes, records, and verifies selected system activities to determine significant departure from expected operation.

Multi-Volume Cartridge (MVC) A physical tape cartridge residing in an LSM that either contains migrated virtual tape volumes (VTVs) or is identified as a volume that can be selected for VTV stacking.

MVCPool Statement An HSC control statement that is contained in the definition data set specified by the VT MVCDEF command. An MVCPool statement specifies the MVCs that VTCS uses.

MVCDEF An HSC command that is used to load the definition data set that contains MVCPool statements.

N

O

output stack The part of the cartridge loader that receives and holds processed cartridges.

P

paired–CAP mode The two forty–cell CAPs in an enhanced CAP function in paired–CAP mode as a single eighty–cell CAP.

PARMLIB control statements Parameter library (PARMLIB) control statements allow you statically specify various operation parameters which take effect at HSC initialization. Identifying your system requirements and then specifying the appropriate control statements permits you to customize the HSC to your data center.

Pass–Thru Port (PTP) A mechanism that allows a cartridge to be passed from one LSM to another in a multiple LSM ACS.

physical end of tape A point on the tape beyond which the tape is not permitted to move.

positional parameter In command and utility syntax, operands that are identified by their position in the command string rather than by keywords (*see* “keyword parameter”). Positional parameters must be entered in the order shown in the syntax diagram.

POST *See* Program for Online System Testing.

PowderHorn A high–performance LSM (model number 9310) featuring a high–speed robot. The PowderHorn has a capacity of up to approximately 6000 cartridges.

Primary. One of two VTSSs in a cluster which is designated in CONFIG as the primary. During normal operations the primary services the host workload and copies replicate VTVs to the secondary.

Program for Online System Testing (POST) A program in a host computer that allows it to test an attached subsystem while the subsystem is online.

Program Temporary Fix A unit of corrective maintenance delivered to a customer to repair a

defect in a product, or a means of packaging a Small Programming Enhancement (SPE).

Program Update Tape A tape containing a collection of PTFs. PUTs are shipped to customers on a regular basis under the conditions of the customer’s maintenance license.

PTF *See* Program Temporary Fix.

PTP *See* pass–thru port.

PUT *See* Program Update Tape.

R

RACF *See* Resource Access Control Facility.

Real Tape Drive (RTD) The physical transport attached to the LSM. The transport has a data path to a VTSS and may optionally have a data path to MVS or to another VTSS.

RDC *See* Remote Diagnostic Center.

recall The movement of VTVs from the MVC back to the VTSS. May be automatic or on demand.

reclaim Refers to MVC space reclamation. For automatic and demand reclamation, VTCS uses the amount of fragmented free space on the MVC and the amount of VTV data that would have to be moved to determine if space reclamation is justified.

Reconciliation. An automatic process initiated when a cluster is reestablished after the primary or secondary has been offline. Reconciliation ensures that the contents of the primary and secondary are identical with respect to replicate VTVs.

Recording Density The number of bits in a single linear track measured per unit of length of the recording medium.

Remote Diagnostic Center (RDC) The Remote Diagnostic Center at StorageTek. RDC operators can access and test StorageTek systems and software, through telecommunications lines, from remote customer installations. Also referred to as the Central Support Remote Center (CSRC).

Replication. Copying a replicate VTV from the primary VTSS to the secondary VTSS in a cluster. When replication completes, there are two copies of

the VTV, one in the primary and one in the secondary.

Replicate VTV. A VTV which has had the replicate attribute attached to it by a management class statement.

Resource Access Control Facility (RACF)
Security software controlling access to data sets.

RTD *See* real tape drive.

S

SCP *See* System Control Program.

scratch tape subpool A defined subset of all scratch tapes. Subpools are composed of one or more ranges of VOLSERs with similar physical characteristics (type of volume {reel or cartridge}, reel size, length, physical location, etc.). Some installations may also subdivide their scratch pools by other characteristics, such as label type (AL, SL, NSL, NL). The purpose of subpooling is to ensure that certain data sets are built only within particular ranges of volumes (for whatever reason the user desires). If a volume which does not belong to the required subpool is mounted for a particular data set, it is dismounted and the mount reissued.

Secondary. One of two VTSSs in a cluster which is designated in CONFIG as the secondary. During normal operations the secondary receives copies of replicate VTVs, stores them, and makes a migration copy on an MVC as soon as possible.

secondary recording A technique for recovery involving maintaining both a control data set and a copy (secondary) of the control data set.

SER Software Enhancement Request.

ServiceTek (machine initiated maintenance) A unique feature of the ACS in which an expert system monitors conditions and performance of subsystems and requests operator attention before a potential problem impacts operations. Customers can set maintenance threshold levels.

servo A device that uses feedback from a sensing element to control mechanical motion.

Small Programming Enhancement (SPE) A supplement to a released program that can affect several products or components.

SMF System Management Facility. An MVS facility used to record system actions which affect system functionality.

SMP System Modification Program.

SMP/E System Modification Program Extended.

SMS System Managed Storage.

SPE Small Programming Enhancement.

standard CAP A standard CAP has a capacity of twenty-one cartridges (three rows of seven cells each). An LSM access door with a standard CAP contains cell locations for storing cartridges. (*see also*, Cartridge Access Port (CAP), enhanced CAP.)

standard LSM A model 4410 LSM which has a storage capacity of up to approximately 6000 cartridges.

standby The status of a station that has been varied online but is connected to the standby LMU of a dual LMU ACS.

standby LMU The redundant LMU in a dual LMU configuration that is ready to take over in case of a master LMU failure or when the operator issues the SWitch command.

station A hardware path between the host computer and an LMU over which the HSC and LMU send control information.

storage class A named list of storage attributes that identify performance goals and availability requirements for a data set. Note that SMS Storage Classes are different from VSM Storage Classes.

storage group A collection of storage volumes and attributes defined by the storage administrator. Note that this is an SMS concept, not a VSM concept.

switchover The assumption of master LMU functionality by the standby LMU.

System Control Program The general term to describe a program which controls access to system resources, and allocates those resources among executing tasks.

system-managed storage Storage that is managed by the Storage Management Subsystem, which attempts to deliver required services for availability, performance, space, and security applications.

System Modification Program Extended An IBM-licensed program used to install software and software maintenance.

T

tape cartridge A container holding magnetic tape that can be processed without separating it from the container.

tape drive A device that is used for moving magnetic tape and includes the mechanisms for writing and reading data to and from the tape.

TAPEREQ An HSC control statement that is contained in the definition data set specified by the TREQDEF command. A TAPEREQ statement defines a specific tape request. It is divided into two parts, the input: job name, step name, program name, data set name, expiration date or retention period, and an indication for specific requests or nonspecific (scratch) requests; and the output: media type and recording technique capabilities. You can use TAPEREQ statements to direct data sets to VSM.

tape unit A device that contains tape drives and their associated power supplies and electronics.

Timberwolf (9740) LSM A high performance LSM that provides a storage capacity of up to 494 cartridges. Up to 10 drives (STD, 4490, 9490, 9490EE, 9840, and SD-3) can be configured. Timberwolf LSMs can only attach to other Timberwolves.

TMS Tape Management System.

TP Tape-to-Print.

transaction A short series of actions with the control data set. These actions are usually related to a specific function (e.g., Mount, ENter).

transport An electromechanical device capable of threading tape from a cartridge, moving the tape across a read/write head, and writing data onto or reading data from the tape.

TREQDEF An HSC command that is used to load the definition data set that contains TAPEREQ control statements.

Tri-Optic label An external label attached to the spine of a cartridge that is both human and machine readable.

TT Tape-to-Tape.

U

UNITATTR An HSC control statement that is contained in the definition data set specified by the UNITDEF command. A UNITATTR statement defines to the HSC the transport's media type and recording technique capabilities. For VSM, the UNITATTR statements define the VTD addresses to VSM as virtual and associate them with a VTSS.

UNITDEF An HSC command that is used to load the definition data set that contains UNITATTR control statements.

utilities Utility programs. The programs that allow an operator to manage the resources of the library and to monitor overall library performance.

V

Virtual Storage Manager (VSM) A storage solution that virtualizes volumes and transports in a VTSS buffer in order to improve media and transport use. The hardware includes VTSS, which is the DASD buffer, and RTDs. The software includes VTCS, an HSC-based host software, and VTSS microcode.

Virtual Tape Control System (VTCS) The primary host code that controls activity and information about VTSSs, VTVs, RTDs, and MVCs.

Virtual Tape Drive (VTD) An emulation of a physical transport in the VTSS that looks like a physical tape transport to MVS. The data written to a VTD is really being written to DASD. The VTSS has 64 or 256 VTDs that do virtual mounts of VTVs.

Virtual Tape Storage Subsystem (VTSS) The DASD buffer containing virtual volumes (VTVs) and virtual drives (VTDs). The VTSS is a STK RAID 6 hardware device with microcode that enables transport emulation. The RAID device can

read and write “tape” data from/to disk, and can read and write the data from/to an RTD.

Virtual Tape Volume (VTV) A portion of the DASD buffer that appears to the operating system as a real tape volume. Data is written to and read from the VTV, and the VTV can be migrated to and recalled from real tape.

virtual thumbwheel An HSC feature that allows read-only access to a volume that is not physically write-protected.

VOLATTR An HSC control statement that is contained in the definition data set specified by the VOLDEF command. A VOLATTR statement defines to the HSC the media type and recording technique of the specified volumes. For VSM, the VOLATTR statements define the volsers for volumes that will be used as MVCs.

VOLDEF An HSC command that is used to load the definition data set that contains VOLATTR control statements.

VOLSER A six-character alphanumeric label used to identify a tape volume.

volume A data carrier that is mounted or demounted as a unit. (*See* cartridge).

VSM *See* Virtual Storage Manager.

VTCS *See* Virtual Tape Control System.

VTD *See* virtual tape drive.

W

WolfCreek A smaller capacity high-performance LSM. WolfCreek LSMs are available in 500, 750, and 1000 cartridge capacities (model numbers 9360-050, 9360-075, and 9360-100 respectively). WolfCreek LSMs can be connected by pass-thru ports to 4410, 9310, or other WolfCreek LSMs.

WolfCreek CAP The standard WolfCreek CAP contains a 20-cell magazine-style CAP and a priority CAP (PCAP). (*see also*, Cartridge Access Port (CAP), Enhanced CAP, standard CAP, WolfCreek optional CAP.)

WolfCreek optional CAP The WolfCreek optional CAP contains a 30-cell magazine-style CAP which

is added to the standard WolfCreek CAP. (*see also*, Cartridge Access Port (CAP), Enhanced CAP, standard CAP, WolfCreek CAP.)

Write Tape Mark (WTM) The operation performed to record a special magnetic mark on tape. The mark identifies a specific location on the tape.

WTM *See* Write Tape Mark.

WTO Write-to-Operator.

WTOR Write-to-Operator with reply.

Symbols

υ **-software.** Microprogram. A sequence of microinstructions used to perform preplanned functions and implement machine instructions.

Numerics

4410 LSM *See* standard LSM.

9310 LSM *See* Powderhorn LSM.

9360 LSM *See* Wolfcreek *LSM*.

9490 Cartridge Subsystem Cartridge tape transports that provide read/write capability for 36-track recording format and extended capacity tape and provide improved performance over the 4490 Cartridge Subsystem. 9490 transports can also read data recorded in 18-track format. The StorageTek 9490 Cartridge Subsystem offers better performance (faster data transfer rate, faster load/unload) than a 3490E device.

9490EE Cartridge Subsystem A high performance tape transport that provides read/write capability for Extended Enhanced (EEtape) cartridges. It is functionally equivalent to the IBM 3490E device.

9740 LSM *See* Timberwolf *LSM*.

9840 Cartridge Subsystem A high performance tape transport system for Enterprise and Open Systems environments that reads and writes 9840 cartridges. 9840s can be defined in 10-drive and 20-drive panel configurations. The 9840 can perform as a stand-alone subsystem with a cartridge scratch loader installed, or it can be attached to a StorageTek ACS.

Index

A

ACSs, duplexing to separate, 47

AMTs

 policies, 41

C

CDS

 formatting, 70, 72

 planning for, 38

CONFIG utility, 10, 17, 19, 23

Configuring VSM

 starting VTCS, 96

configuring VSM, 89

connecting MVS/CSC clients to VSM, 84

D

definition data sets for VSM, 9, 73

duplexing VTVs, 46

E

esoterics

 configuring, 54

 JES2 and JES3, 95

 planning for, 12

ExLM SYNCVTV function, 42

H

HCD facility, 11

high AMT

 policies, 41

hosts

 enabled for migration, 43

HSC

 defining a security system user ID, 54

 reconfiguration

 creating an MVC pool, 24, 73

 formatting the new CDS, 70, 72

 LIBGEN definitions for RTDs, 19

 MVC VOLATTR statements, 73

 MVCPool statements, 73

 overview, 67

PARMLIB member, updating for VSM, 18, 82

RTD UNITATTR statements, 73

updating definition data sets for VSM, 73

updating the LIBGEN, 68

verifying the LIBGEN, 70

VTD UNITATTR statements, 73

SLUADMIN utility, 91

user exits

 SLSUX02, 95

 SLSUX04, 95

 SLSUX15, 91

I

installation

 planning, 1

 summary and checklist, 2

J

JES2 environment

 esoterics, 12

 user exit SLSUX02, 95

JES3 environment

 esoterics, 12

 user exit SLSUX04, 95

L

LIBGEN

 updating, 68

 verifying, 70

low AMT

 policies, 41

M

migration

 policies, 39

MIH, setting, 55

MVCPool control statement, 24, 73

MVCs

 creating an MVC pool, 24, 73

 defining and selecting, 22

 defining available with CONFIG utility, 23

 managing scratch status, 25

- maximum retain interval, 46
- maximum VTVs per, 46
- planning for, 20
- policies
 - migration, 46
 - space reclamation, 50
- protecting, 25
- RTV access, 91
- space reclamation
 - policies, 49, 50, 51, 52
- tape management system, 25
- unscratching current scratch cartridges, 91
- VOLATTR control statement, 73

P

PARMLIB member for VSM, 82

R

- Reconfiguring HSC
 - restarting HSC, 87
- recovery utility
 - MVC access, 91
 - VTV access, 91
- routing data sets to VSM, 93
- RTDs
 - defining for sharing with MVS, 19
 - LIBGEN definitions, 19
 - planning for, 19
 - policies, 46
 - UNITATTR control statement, 19, 73
- RTV utility
 - MVC access, 91
 - VTV access, 91

S

- scratch subpools
 - for VTVs, 18
- security
 - defining for VSM, 90, 91
 - planning for, 25
- SLSUX15 user exit
 - issuing command authorization request to security product, 91
- SLUADMIN utility
 - unscratching current scratch cartridges, 91
- SMC/DFSMS interface, 12
- SMF record formats, 83
- SMFPRMxx, producing SMF records, 83
- space reclamation

MVCs, 49, 50, 51, 52

T

- tape management system
 - planning for, 17, 25, 38
 - updating, 92
- TAPEREQ control statement, 12
- TAPEVOL security class, 90

U

UNITATTR control statement, 10, 19, 73

V

- VOLATTR control statement, 73
 - examples, 20, 21
- VSM
 - candidate data sets, 38
 - configuration
 - defining VSM security, 90, 91
 - record, 97
 - routing data sets to VSM, 93
 - updating the tape management system, 92
 - values, 9
 - installation
 - planning, 1
 - summary and checklist, 2
 - Nearline hardware requirements, 7, 8
 - online documentation, xv
 - policies
 - AMTs, 41
 - duplexing, 46
 - hosts enabled for migration, 43
 - maximum concurrent migration tasks, 39
 - maximum VTVs per MVC, 46
 - migration, 43, 45, 46, 47
 - MVC space reclamation, 52
 - RTDs, 46
 - space reclamation, 50
 - related publications, xi
 - security, 91
 - software and hardware prerequisites, 4
 - StorageTek technical support, xv
- VSM, configuring, 89
- VTCS
 - configuration values, 9
 - installation
 - APF authorizing the VTCS LINKLIB, 65
 - applying the VTCS and HSC base FMIDs, 63
 - applying the VTCS service, 65

- creating a VTCS LINKLIB data set, 62
- defining libraries to the HSC target zone, 62
- installing the HSC base FMID and maintenance, 64
- modifying the HSC startup procedure, 65
- overview, 57
- planning, 1
- receiving the VTCS FMID, 61
- receiving the VTCS service, 61
- summary and checklist, 2
- verifying installation materials, 58
- online documentation, xv
- pre-installation
 - configuring MVS unit addresses/esoterics, 54
 - defining a security system user ID, 54
 - setting the MIH value, 55
- publications, xi
- related publications, xi
- SMF record formats, 83
- VTDs
 - configuring
 - esoterics, 54
 - MVS device numbers, 54
 - data chaining a read forward or write command, 38
 - MVS device numbers, 11
 - planning for, 11
 - UNITATTR control statement, 10, 73
- VTSSs
 - planning for, 10
- VTVs
 - defining to tape management system, 17
 - duplexing
 - policies, 46
 - maximum per MVC, 46
 - migration
 - on dismount, 45
 - policies, 39, 43, 45, 47
 - planning for, 17
 - policies, 46
 - RTV access, 91
 - scratch subpools, 18

